Prof. Dr.-Ing. Matthias L. Hemmje

63413
Dokumenten- und Wissensmanagement im Internet

Lektion 1: Das Modell der strukturierten Dokumente

Leseprobe

Fakultät für
Mathematik und Informatik

FernUniversität in Hagen

Das Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere das Recht der Vervielfältigung und Verbreitung sowie der Übersetzung und des Nachdrucks, verbleiben bei der FernUniversität. Das Werk darf gemäß § 53 UrhG im privaten Rahmen zum Beispiel durch Kopien oder digitale Speicherung vervielfältigt werden.

Inhaltsverzeichnis

Inhaltsverzeichnis III
1. Einleitung 4
2. Der Dokumentenbegriff 4
3. Der Modellbegriff 5
4. Realisierung des Dokumentenmodells 5
5. Zeichencodierungen am Beispiel Unicode 24
6. Zusammenfassung und Ausblick 32
7. Anhang: Formate am Beispiel 33
8. Literaturverzeichnis 40
9. Abbildungsverzeichnis 42
10. Tabellenverzeichnis 42
11. Lösungen der Selbsttestaufgaben 43

1. Einleitung

Dies ist die erste Lektion der Lehrveranstaltung 63413 „Daten-, Dokumenten-, Informations- und Wissensmanagement im Internet".

In diesem Abschnitt werden allgemeine Formate für Dokumente sowie für strukturierte Dokumente, die im Internet verfügbar sind, vorgestellt. Die Teilnehmerinnen und Teilnehmer setzen sich zunächst mit einer Präzisierung des Dokumentenbegriffs und dem des Modells auseinander.

Es folgt eine detaillierte Betrachtung des Modells der strukturierten Dokumente, in der die Studierenden verschiedene Arten von Textdokumenten, Aspekte der Strukturierung sowie Umsetzungsmöglichkeiten des Modells analysieren.

Weiter untersuchen sie verschiedene Modelle von Zeichencodierungen, die den Inhalt von Dokumenten in maschinenlesbarer Form repräsentieren, sowie deren einzelne Bestandteile, beispielsweise Codierungsschema und Übertragungssyntax.

Die vorliegende Lehrunterlage versteht sich als überblicksartiger Einstieg in die Thematik. Ein eigenständiges Studium der aktuellen Sprachstandards für HTML, CSS und Java Scripts anhand der offiziellen Veröffentlichungen (siehe Links) ist erforderlich.

2. Der Dokumentenbegriff

Für statisch-passive Dokumente wurde bereits in den achtziger Jahren das Modell der strukturierten Dokumente entwickelt. Durch dieses Modell wird eine Unterteilung in textuellen Inhalt, logische Struktur und grafisches Format möglich. Die Quellrepräsentation des Dokuments bezieht sich auf den textuellen Inhalt und logische Struktur. Gesteuert von unabhängigen Formatvorgaben (engl. *stylesheets*) lässt sich daraus das grafische Format generieren.

Quellrepräsentation

Moderne Systeme zur Erstellung und Verwaltung von Dokumenten und großen Dokumentenbeständen profitieren vom Modell der strukturierten Dokumente, da es ein hohes Maß an Flexibilität bietet. Die logische Strukturierung unterstützt Anwendungen der Dokumentenverwaltung wie die automatische Generierung von Katalogeinträgen und Verzeichnissen oder die Wiedergewinnung von Information (engl. *information retrieval*) nach bestimmten logischen Kriterien und erlaubt es ganz allgemein, Dokumente als Daten zu betrachten, die mithilfe von Computersystemen verwaltet und manipuliert werden können.

Stylesheets

Die Unabhängigkeit der Stylesheets von den Dokumenten ermöglicht auf der einen Seite, dass ein- und dasselbe Dokument durch Kombination mit verschiedenen Stylesheets passend zum Anwendungszweck formatiert werden kann; auf der

anderen Seite kann man verschiedene Dokumente durch Kombination mit ein- und demselben Stylesheet einheitlich formatieren. Auch lässt sich die Struktur beim Erstellen und Lesen von Dokumenten nutzen, etwa zur Navigation und zur Definition von Sichten etwa von Outline-Sichten.

3. Der Modellbegriff

Modelle sind ein wichtiges, methodisches Mittel. Die Modellierung ist eine vereinfachte und auf das im aktuellen Kontext Wesentliche reduzierte und – unter geeigneten Gesichtspunkten – systematisierte Darstellung. Modelle sind aus vielen Bereichen bekannt wie etwa aus der Architektur, der Biologie oder der Physik. Modelle können physikalischer oder gedanklicher Art sein; letztere können umgangssprachlich oder formal definiert sein. Gegenstandsbereich einer Modellierung können Konzepte, Vorgänge oder Erscheinungen und ihre Wechselwirkungen sein. Modelle können unterschiedlichen Zwecken dienen: etwa der Entwicklung einer gemeinsamen Sicht und Terminologie, der Vermittlung erprobter Vorgehensweisen und Techniken, dem rationalen Beweisen oder Widerlegen von Hypothesen oder der Ableitung von Handlungsempfehlungen.

Zwei zentrale und umgangssprachlich erläuterte Modelle werden in dieser Lektion eingeführt und näher erläutert: das Modell der strukturierten Dokumente und ein Modell für Zeichencodierungen.

4. Realisierung des Dokumentenmodells

Hier wird anhand verschiedener Beispiele und Systeme erläutert, wie das Modell der strukturierten Dokumente mit Zeichencodierungen, eingebetteten Auszeichnungen (Markierungen, engl. *markup*), Dokumentengrammatiken und Stylesheets in der Praxis umgesetzt wird. Auf der technischen Seite wird das Thema der Zeichencodierungen vertieft. In den nachfolgenden Lektionen wird das Thema der strukturierten Dokumente anhand gängiger Internettechnologien weiter beleuchtet. Dazu gehören XML und die zugehörigen weiteren XML-basierten Sprachfamilien.

Dokumente näher betrachtet

Historisch gesehen findet sich beispielsweise in Brockhaus' Konversationslexikon in der vierzehnten Auflage von 1894 folgendes unter dem Begriff „Dokument":

ursprünglicher Dokumentenbegriff

„Im weiteren Sinne jeder Gegenstand, welcher dazu dient, die Wahrheit einer zu erweisenden Tatsache, besonders einer für ein Rechtsverhältnis erheblichen Tatsache, zu bestätigen. Im engeren Sinne versteht man darunter Urkunden oder Schriftstücke, im Gegensatz zu anderen körperlichen Beweisstücken, wie Grenzsteinen, Wappen, beschädigten Sachen."

Ursprünglich waren Dokumente also tatsächlich physische Objekte, Gegenstände, die als einmalige Träger eines in menschliche Sprache gekleideten und schriftlich festge-

haltenen Gedankens oder Sachverhalts fungierten. Ein Dokument neuerer Art, etwa ein sauber getippter Bogen Papier mit Name, Adresse, Personalausweisnummer und weiterer Information zur Person erfüllt einfach nicht denselben Zweck, auch wenn er dieselbe sprachliche Information enthält. Die Dauerhaftigkeit der Schriftform gegenüber der gesprochenen Sprache, die Kommunikation unabhängig von Zeit und Raum ermöglicht, und die Authentizität des Trägermediums, die durch Wasserzeichen, Stempel, Siegel, Barcode oder Ähnliches erreicht werden kann, machen das Wesen von Dokumenten in diesem ursprünglichen Sinne aus.

In der Brockhaus-Enzyklopädie in der siebzehnten Auflage von 1968 findet sich dagegen unter dem Stichwort „Dokument":

neuerer Dokumentenbegriff

> *„Als Dokumente können alle Unterlagen betrachtet werden, die Information beinhalten, also nicht nur publiziertes Wissen, sondern auch Briefe, Akten, Urkunden, Bildsammlungen, Filme u.v.a."*.

Die Dokumente neuerer Art sind also Informationsträger. Die Schriftform ist immer noch wesentlich, aber das Trägermedium und die Art, wie die Information auf das Trägermedium aufgebracht ist, sind nebensächlich. Auf den sprachlichen Ausdruck kommt es an.

Der Unterschied zwischen dem ursprünglichen und dem neueren Dokumentenbegriff kann an zwei Arten von Delikten klargemacht werden. Das Fälschen eines Personalausweises oder einer Banknote beinhaltet immer auch eine Fälschung des Trägermediums und betrifft Dokumente im ursprünglichen Sinne. Kopieren hingegen – sei es Fotokopieren oder Abschreiben – generiert immer ein zweites Exemplar des Dokuments, selbst wenn sich die Erscheinungsform dabei total ändert. Die Umstände, unter denen ein solches Kopieren zulässig ist, regelt das Urhebergesetz.

Das Gemeinsame an Dokumenten ursprünglicher und neuerer Art ist, dass sie durch ihre Schriftform eine ursprünglich sprachliche Mitteilung von der auditiven in die visuelle Übertragungsform und von der Flüchtigkeit der mündlichen Erzählung in die Dauerhaftigkeit des geschriebenen Wortes umsetzen. Somit werden die Grenzen von Zeit und Raum, die sonst der Senderin und dem Empfänger flüchtiger sprachlicher Mitteilungen gesetzt sind, überschritten.

Dokumente ursprünglicher und neuerer Art unterscheiden sich auch dadurch, wie viele Exemplare es von einem Dokument geben kann und wie viele Leserinnen und Leser gleichzeitig Gebrauch von so einem Dokument machen können.

Eine weitere und wichtige Unterscheidung, die zu einem dritten, dem modernen Begriff des digitalen Dokuments führt, hängt an der Idee vom Dokument als Informationsträger. Sprachlich formulierte Gedankengänge, Ideen, Fakten oder Aussagen werden erst dann zu Information, wenn sie von jemandem benötigt, aufgenommen und verarbeitet werden. Im folgenden Abschnitt, der auf Kuhlen [Kuh89] basiert, wird dies beleuchtet:

Information ist Wissen in Aktion

„Information ist Wissen in Aktion" lautet eine Kurzformel der Informationswissenschaft, die sinngemäß von R. Kuh eingeführt wurde: "Information [..] as knowledge in action".

Er unterscheidet dabei die Konzepte von Information und Wissen dahingehend, dass Information als Teilmenge von Wissen aufgefasst wird. Weiterhin definiert er, dass Information zur Lösung eines konkreten Problems durch eine Person benötigt wird, aber nicht verfügbar ist. Dabei geht Kuhlen nicht auf den Grund für die Nicht-Verfügbarkeit der Information ein, im Gegensatz z. B. zu P. Wilson [Wil95], der erläutert, wie Informations-Überflutung zum „Nonuse" von Information führt. Information als Wissen in Aktion bildet laut Kuhlen frei übersetzt das „pragmatische Primat der Informationsverarbeitung" („pragmatic primacy of Information Work".

Für die Informationsverarbeitung und das Design von Informationssystemen zieht er daraus die Schlussfolgerung, dass die Hauptaufgabe jeglicher Informationsverarbeitung darin besteht, Wissen in [nutzbare] Information zu transformieren und dabei Randbedingungen wie Zeitbedarf, Kosten, soziale Aspekte, kognitive Fähigkeiten und Ziele der jeweiligen Organisation zu berücksichtigen.

Weiterhin forderte Kuhlen, das pragmatic primacy of Information Work bei zukünftigen Entwicklungen von Informationssystemen ernst zu nehmen und zu berücksichtigen.

Mit Blick auf den Dokumentenbegriff bedeutet der Informations- und Wissensbegriff im Verständnis von Kuhlen, dass im Dokumentenbegriff Wissen beinhaltet ist. Dieses Wissen wird für einen Menschen erst in dem Moment zu Information, in dem er es z. B. für die Bearbeitung einer Aufgabe oder die Lösung eines Problems benötigt. Und auch das nur insoweit, als es nicht bereits Bestandteil seines menschlichen Wissens ist.

Dieses Bedürfnis nach Wissen wird in der Informationswissenschaft als Informationsbedürfnis bezeichnet. Das bedeutet weiterhin, dass es in der maschinellen Verarbeitung von Dokumenten nicht nur darauf ankommt, dort Wissen zu codieren, sondern dass es auch darauf ankommt, Dokumentinhalte nach Wissen durchsuchbar zu machen, welches für ein real existierendes menschliches Informationsbedürfnis relevant und damit Information im Sinne von Kuhlen ist.

Was bedeutet das konkret? In der Natur der Dinge liegt es dabei, zunächst an ein menschliches Gegenüber zu denken. Wie verhält es sich jedoch mit Computern, die ja dazu gedacht sind Daten, Dokumente, und somit implizit in diesen enthaltenes Wissen, und darin wiederum ggf. enthaltene Information entgegenzunehmen, zu durchsuchen, weiterzuverarbeiten und anschließend verändert wieder auszugeben, wie dies ja bereits durch das EVA-Prinzip definiert wird?

Daten-, Dokumenten- und Informationsverarbeitung mithilfe von Computerprogrammen führt demnach unmittelbar zum modernen Dokumentenbegriff.

Wesentliche Tätigkeiten im Bereich der modernen Informationsverarbeitung werden heute durch Computersysteme unterstützt. Dies hat dazu beigetragen den modernen Dokumentenbegriff zu prägen. Unterstützt werden dabei Tätigkeiten und Anwendungen wie:

- Gedankengänge organisieren und Dokumente planen,
- Gedanken verbalisieren und Dokumente erfassen,
- Dokumente editieren, formatieren und publizieren, also der Öffentlichkeit zugänglich machen,
- Dokumente ablegen, verwalten, abfragen, zusammenfassen, kombinieren, transformieren, extrahieren,
- Dokumente lesen, verstehen, memorieren, aus ihnen lernen und über sie nachdenken.

der moderne Dokumentenbegriff

Moderne Dokumente sind Informationsträger, die alle diese Anwendungsszenarien effizient unterstützen. Moderne Dokumente müssen deshalb in ihrer Urform elektronisch gespeichert sein und je nach Anwendungszweck unterschiedliche Präsentationsmedien unterstützen.

Selbsttestaufgabe 1.1:

Erläutern Sie, wie von einer Suchmaschine Kuhlens „Primacy of Information Work" bei der Suche nach dem Stichwort „NASA Perseverance" umgesetzt wird. Die Suche wird von einem Vater ausgeführt, um Fragen seiner Tochter Melissa zu beantworten.

Selbsttestaufgabe 1.2:

Überlegen Sie, ob auf die folgenden Dokumente eher der ursprüngliche, der neuere oder der moderne Dokumentenbegriff passt oder ob es sich vielleicht gar nicht um Dokumente handelt:

- Personalausweis
- Geldschein
- Gemälde
- Veranstaltungsplakat
- Privatbrief
- E-Mail-Nachricht
- Memo an die Geschäftsleitung
- Gedichtband
- Roman
- Bestellung
- Produktbeschreibung
- Fragebogen
- Nachricht in einem Protokoll zum digitalen Zahlungsverkehr (Electronic Banking)

Nicht alle Beispiele lassen sich eindeutig einordnen. Machen Sie sich klar, welche Argumente für oder gegen eine bestimmte Einordnung sprechen.

Vorangehend wurde eine Übersicht über die wesentlichen Kategorien von Dokumentenbegriffen gegeben. Im weiteren Verlauf werden in dieser Lehrveranstaltung jedoch nur noch Dokumente der modernen Art besprochen.

Dokumente sind also nachfolgend immer elektronisch gespeicherte Wissens- und ggf. auch Informationsträger, deren schriftsprachlich festgehaltene Inhalte sowohl von menschlichen Leserinnen und Lesern als auch von Computerprogrammen wahrgenommen und weiterverarbeitet werden können.

Im folgenden Abschnitt, der auf [BOB82] basiert, wird noch kurz der Aspekt des Informationsbedürfnisses und der Informationssuche angesprochen, bevor sich dann eine Analyse anschließt, welche Anforderungen sich daraus für Dokumentenformate ergeben und wie diese Anforderungen durch das Modell der strukturierten Dokumente erfüllt werden.

Verwandt zum Informations- und Wissenskonzept von Kuhlen ist das Konzept des Anomalous State of Knowledge (ASK) nach Belkin, welches folgendermaßen von ([BOB82], p. 62) definiert wird:

> *"The ASK hypothesis is that an information need arises from a recognized anomaly in the user's state of knowledge concerning some topic or situation and that, in general, the user is unable to specify precisely what is needed to resolve that anomaly."*

Das bedeutet, unter einem ASK versteht „man einen „anormalen" Wissenszustand, womit gemeint ist, dass einem Menschen zum Beispiel für die Bearbeitung einer Aufgabe oder zur Lösung eines Problems Wissen fehlt.

Unter einem „normalen" Wissenszustand wird dabei verstanden, dass der Mensch das benötigte Wissen bereits besitzt. Insofern ist der ASK nichts anderes als die akademische Beschreibung eines Informationsbedürfnisses eines Menschen. Dies wird dann weiter so ausgeführt:

Ein ASK kann zu einem Informationsbedürfnis („Information Need") führen, welches beispielsweise von [MRS08] wie folgt definiert wird:

> *„An information need is the topic about which the user desires to know more, and is differentiated from a query, which is what the user conveys to the computer in an attempt to communicate the information need".*

Das aus einem ASK resultierende Informationsbedürfnis stellt somit eine wichtige Anwendung der automatisierten Dokumentverarbeitung dar. Computersysteme, die durch automatische Dokumentverarbeitung Informationsbedürfnisse erfüllen können,

indem sie relevantes Wissen in den Dokumenten suchen, heißen Informationssysteme. Umgangssprachlich werden sie oft als Suchmaschinen bezeichnet.

Diese implizite Beschreibung einer Interaktion eines Nutzenden mit einer Suchmaschine während des Versuchs das ASK zu formulieren, wird als „Informational Behaviour" bezeichnet. Laut Bates [Bat10, p. 2078] besteht Informational Behaviour aus der Interaktion von Nutzenden mit Informationssystemen, der Frage, wie und wann sie Information suchen und was die Nutzenden daraus machen.

Weiterhin stellt sie fest, dass der Vorgang des Suchens an sich ein interessantes Forschungsgebiet darstellt. In diesem Forschungsgebiet wurden daher verschiedene Suchstrategien („Information Seeking", „Information Searching") beschrieben. Im Rahmen einer Studie stellte Bates fest, dass die Strategien der Nutzenden heuristisch und hochgradig interaktiv, gleichzeitig aber wenig geplant waren. Dabei unterscheidet sie zwei grundlegende Suchstrategien, nämlich „Lookup" (Nutzung der Textsuche einer Suchmaschine) und „Examine" (das Anzeigen von Titeln und Texten). Es wird dargelegt, dass Suchmaschinen neben Schlüsselwort-basierter Suche auch Browsing-orientierte Suchstrategien („Berrypicking") unterstützen sollen. Die verschiedenen Modi der Informationssuche sind in Abbildung 4.1 zusammengefasst.

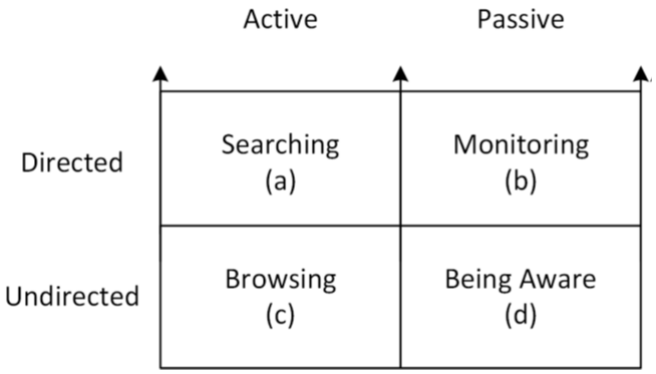


Abbildung 4.1 Modi der Informationssuche [Bat02]

Dabei wird zwischen aktiver und passiver sowie gerichteter und ungerichteter Suche unterschieden. Entlang dieser Dimensionen unterscheidet Bates:

- direkte Suche als aktive Versuche, Fragen zu einem speziellen Thema zu beantworten und das Informationsbedürfnis zu befriedigen,
- Monitoring als aktiver Aufbau von Hintergrundwissen zu Interessengebieten und möglichen zukünftigen möglichen Fragestellungen,
- Browsing als das Gegenteil von Monitoring, um neues Wissen ohne konkretes Informationsbedürfnis durch aktive Auseinandersetzung mit neuen Inhalten zu erwerben,
- allgemeine Aufmerksamkeit (d) als ungerichtete und passive Form der Informationsbeschaffung. Laut Bates „gives us 80 percent of all we know".

SECI-Modell

Da Dokumente Information und somit auch Wissen beinhalten, wird an dieser Stelle außerdem das sogenannte SECI-Modell betrachtet. Das SECI-Modell wurde von Nonaka eingeführt und später in den 90er Jahren von Takeuchi verfeinert [NT95]. Es beschreibt die Methoden, mit denen implizites Wissen in explizites Wissen umgewandelt wird und umgekehrt.

Das SECI-Modell kann wie folgt beschrieben werden:

1. Sozialisation (implizit zu implizit): Durch den Reifungsprozess sammelt jeder anderes Wissen. Auf diese Weise werden Fähigkeiten und Erfahrungen von Generation zu Generation weitergegeben. In der Vergangenheit, vor der Erfindung von Büchern, Fernsehen oder Internet, war der einzige Weg, Wissen zu übertragen, das Sozialisieren: Menschen kommunizieren direkt mit Menschen oder wenden den Wissensaustausch durch Erfahrungsaustausch an. Heutzutage werden viele fortschrittliche Technologien eingeführt, aber das Sozialisieren bleibt wichtig und kann nicht ersetzt werden.

2. Externalisierung (implizit bis explizit): Geselligkeit erfordert direkte Interaktion zwischen Menschen. Dies ist zeitaufwändig und nicht immer über eine lange Strecke möglich. Zusammen mit der Erfindung des Schreibens kann Wissen vom Gehirn auf andere Medien wie z. B. Bücher, Videos usw. übertragen werden Dieser Vorgang wird als Externalisierung bezeichnet.

3. Kombination (explizit zu explizit): Durch die Transformation von explizitem Wissen in neues explizites Wissen entsteht kein neues Wissen. Dieser Prozess besteht darin, das Wissen zu organisieren und Teile davon in neuen Formen zu kombinieren, damit es sinnvoller ist oder neue Erkenntnisse erklärt.

4. Internalisierung (explizit zu implizit): Personen können neues Wissen erwerben, indem sie explizites Wissen wahrnehmen, d. h., verbrauchen, manchmal, indem sie wiederholt dieselbe Aktivität ausführen. Sie wandeln ihre Erfahrungen und Erinnerungen in mentale Modelle um. Einmal verinnerlicht, kann neues Wissen wiederverwendet, dokumentiert oder an andere Personen weitergegeben werden [Vu15]. Die folgende Abbildung zeigt das Konzept des Modells.

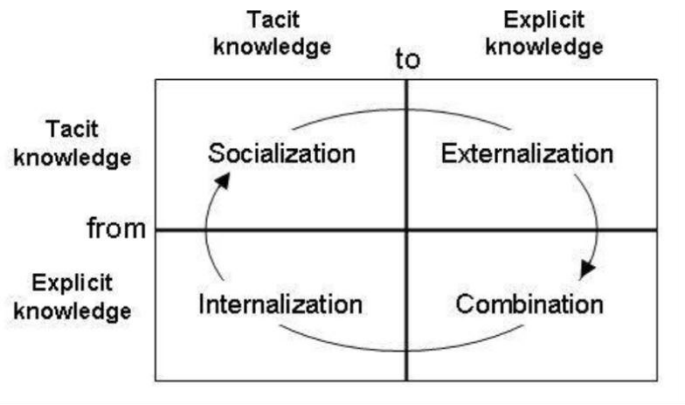


Abbildung 4.2 Das SECI Model [NT95]

Wissen existiert nicht von selbst, sondern wird aus Erfahrungen geschaffen und unter Verwendung verschiedener Arten von Medien oder durch direkte soziale Interaktion zwischen Menschen übertragen. Mit der Erfindung des Personal Computers erhalten Menschen Unterstützung von Computern beim Content Management und damit beim externalisierten Wissensmanagement. Infolgedessen wird dieser Prozess viel effizienter und effektiver als zuvor.

Moderne Textdokumente

Unter dem Begriff „Moderne Textdokumente" werden Textdokumente verstanden, die in weltweit verteilten Informationssystemen wie dem Internet und dem World Wide Web für die unterschiedlichen Nutzungsarten und Anwendungszwecke bereitgestellt werden. Einige Fragen die sich aus diesem Kontext ergeben sind:

- Welche Anforderungen ergeben sich für die Dokumentenformate?
- Welche Formate sind in der Praxis verfügbar?
- Welche Werkzeuge stehen zur Verfügung, um die Dokumente zu bearbeiten?
- Was können etablierende und etablierte Anwendungen wie Document Engineering, Content Management und Elektronisches Publizieren beitragen?

HTML

Zu Beginn der rasanten Entwicklung des WWW gab es nur ein relevantes Dokumentenformat im Web, nämlich die *Hypertext Markup Language*.

IETF
RFC

Die erste konsolidierte, und von der Internet Engineering Task Force verabschiedete, HTML-Version 2.0 umfasste ca. 150 Sprachelemente, wobei mit Sprachelementen sowohl die Elemente als auch die Attribute gemeint sind. Die besagte HTML-Version wurde im November 1995 als RFC 1866[1] eingeführt. Elektronisch arbeitende Produzenten/-innen und Verleger/-innen von Inhalten, sogenannte *Content Provider*, konnten mit einer Handvoll von HTML-Befehlen auskommen, um eine dem aktuellen Stand der Technik entsprechende Webseite zu erstellen.

[1] https://www.rfc-editor.org/rfc/rfc1866

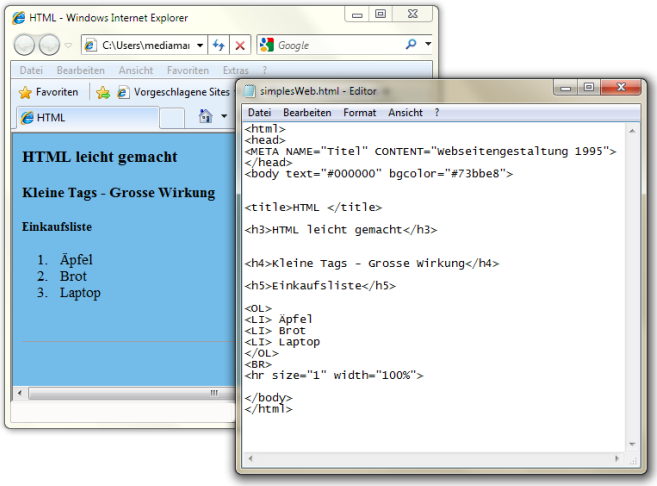


Abbildung 4.3 Webseitengestaltung von 1995

W3C

Die HTML-Version 4.0, welche am 24. April 1998 vom World Wide Web Consortium verabschiedet wurde, umfasste dann bereits ca. 1.600 Sprachelemente. An für Textdokumente relevanten Formaten sind zu HTML die Sprachen

- *E*xtensible *M*arkup *L*anguage,
- *C*ascading *S*tyle *S*heets,
- *E*xtensible *S*tyle *L*anguage,
- XML Schema,
- XML Linking Language,
- XML Path Language,
- XML Pointer Language

hinzugekommen.

Mittlerweile ist die fünfte Fassung von HTML aktuell (HTML5), welche das W3C am 28. Oktober 2014 als fertige Spezifikation vorgelegt hat. Diese ersetzt gleich mehrere Standards. Neben dem umfangreichen Vokabular wird u. a. ein klares Strukturmodell definiert, um modernen Textdokumenten gerecht zu werden. In Lektion 2 wird ausführlicher auf HTML5 eingegangen. Nicht zuletzt ist HTML5 aktuell auch für die Unterstützung von Aspekten wie barrierefreie und mobile Nutzung des Internets zum Daten- und Dokumentenmanagement unverzichtbar.

Neben den statisch-passiven Textdokumenten, die klassischen, auf Papier gedruckten Dokumenten ähnlich sind, haben im Internet bereits seit längerem in zunehmendem Maße dynamisch-aktive Textdokumente an Bedeutung gewonnen. Sie enthalten aktive Komponenten, mit denen Leserinnen und Leser interagieren können. Es können sich auch der Inhalt und das Format dynamisch ändern. Beispiele für aktive Komponenten sind mathematische Formeln, die mittels eines Computer-Algebra-Pakets auf Knopfdruck evaluiert werden können. Weitere Beispiele für dynamische Dokumente sind sich

regelmäßig aktualisierende Börsenkurse oder auf Knopfdruck expandierbare Übersichtsdarstellungen (engl. *outline*).

W3C

Zahl, Umfang und Funktionsbereich der relevanten Empfehlungen des W3C und deren Implementierungen haben bis heute derart zugenommen, dass eine systematische Einführung in die Sprachen selbst und in darauf aufbauende erprobte Praktiken der professionellen und systematischen Dokumenterstellung und -verarbeitung einen organisierten Einstieg in die Internetpraxis der verteilten Informationsverarbeitung sehr erleichtern.

Die neuen Dokumentenformate für statisch-passive Dokumente im Internet folgen ebenfalls dem Modell der strukturierten Dokumente. Neben HTML hat XML für das Daten- und Dokumentenmanagement im Internet in den vergangenen Jahren immer mehr an Bedeutung gewonnen.

Dokumentenbestandteile

Dokumente sind als Informationsträger zu sehen. Wie aber kommen die Leserinnen und Leser an die in einem Dokument enthaltene Information? Als Einstieg in Beantwortung dieser Frage kann zunächst einmal ein konkretes Dokument in Abschnitt 7.a im Anhang betrachtet werden.

Es handelt sich hierbei um eine beispielhafte Beschreibung des Lehrgebiets Multimedia und Internetanwendungen der FernUniversität in Hagen. Wie genau aber wurde diese Information im Dokument codiert, wie wird sie vom Leser wahrgenommen und gedanklich verarbeitet?

Inhaltsebene des Dokumentes

Die Ausgangsbasis, und damit die erste Ebene im Modell der strukturierten Dokumente, sind die in dem Dokument enthaltenen textuellen Inhalte, also die Buchstaben und Interpunktionszeichen, die beim Lesen vom Menschen wie auch von Maschinen zu Wörtern und Sätzen zusammengesetzt werden. Darüber hinaus werden diese von Menschen aber auch intellektuell verarbeitet und mit einer Bedeutung versehen.

Semantik

Die Belegung von Symbolen mit Bedeutungen wird Semantik genannt. Der informationelle Inhalt des Dokuments wird also vom Lesenden erfasst. Dies ist eine Fähigkeit, die der Mensch den Maschinen zunächst voraushat. In der letzten Phase der Erfassung gibt es also wesentliche Unterschiede zwischen der Art und Weise wie ein Mensch und wie eine Maschine Inhalte wahrnimmt, deren informationellen Gehalt erfasst und verarbeitet.

Es sind also zunächst nicht nur die Buchstaben und Interpunktionszeichen alleine, die den Erfassungsprozess ermöglichen. Zu dieser Einsicht verhilft folgender Gegentest: Betrachtet ein menschlicher Lesender die unformatierte Version des gleichen Dokumentes, die sich unter Abschnitt 7.b im Anhang befindet, und versucht, aus dieser Version die Gliederung des Konzepts herauszulesen, so stellt sich dies als mühsam bis unmöglich heraus.

Offenbar unterstützt also die Formatierung des Dokuments – etwa die Wahl der Schriften (groß oder klein, fett oder kursiv), die Ausrichtung von Textpassagen (linksbündig, rechtsbündig oder zentriert) und die Abstände zwischen einzelnen Textpassagen – die visuelle Aufnahme und Verarbeitung der Information durch den Menschen. Geht man zurück zur zuerst betrachteten, formatierten Version des Konzepts, so wird einem menschlichen Lesenden deshalb sofort die Gliederung in die Abschnitte klar. Die visuelle Formatierung der Überschriften in Fettschrift und mit deutlichen Abständen zum umgebenden Text geben ausreichende Hinweise um die Gliederungselemente zu erkennen.

Die Buchstaben, Interpunktionszeichen und sonstigen Symbole, die in einem Dokument enthalten sind, zusammen mit dem Format, also der typografischen, sprich visuellen, Ausprägung der Symbole und der geometrischen Anordnung oder dem Layout, ermöglichen es also den menschlichen Leserinnen und Lesern, den Informationsgehalt eines Dokumentes leichter, d. h. kognitiv effizienter zu erfassen.

Was aber ist genau das Ergebnis dieses menschlichen Verarbeitungsprozesses? Zunächst einmal werden die Aussagen wörtlich erfasst und verstanden. Hinzu kommt jedoch auch das Erfassen der einzelnen Textpassagen und deren Rolle im Gesamttext.

logische Struktur

Zusätzlich zur Aussage eines Dokuments erkennen menschliche Leserinnen und Leser also aufgrund des Inhalts und des Formats eine Strukturierung in logische Einheiten.

Die von den Lesenden erkannte Struktur ist nicht eindeutig. Hinweise auf die Struktur ergeben sich aus Inhalt und Format aufgrund von dem semantischen Verständnis der Textpassagen und der kulturellen, durch den Umgang mit zahlreichen Dokumenten ähnlicher Art gewonnenen Erfahrung.

Für menschliche Lesende stellt sich also der Prozess der Informationsaufnahme und -verarbeitung folgendermaßen dar: Das Dokument wird mit Inhalt und Format präsentiert. Menschliche Leserinnen und Leser extrahieren daraus die Aussage des Dokumentes und konstruieren eine Struktur für die einzelnen Textelemente. Durch visuelle Wahrnehmung und gedankliche Arbeit unter Rückgriff auf das Sprachverständnis und das kulturelle Wissen wird die Bedeutung der Aussage interpretiert. Die Verwendung von Formatmerkmalen, Aussage und Struktur bilden die Grundlage für die weitere intellektuelle Informationsverarbeitung.

Ein Dokument kann in ganz unterschiedlichen grafischen Aufmachungen dargestellt werden. Die Darstellung hängt auch vom Präsentationsmedium und der Lesesituation ab. Beispielsweise kann ein Dokument für das ermüdungsfreie Lesen am Bildschirm mit einer anderen Farbpalette und mit anderen Schriften formatiert sein als für das Lesen von einem Papierausdruck; oder Querverweise können für die Onlineversion als Hypertextlinks und für die Druckversion als Hinweise auf Seitenzahlen realisiert werden.

Entscheidend für die „Identität" eines Dokumentes ist, dass aus den verschiedenen Formaten immer noch dieselbe Aussage und im Wesentlichen dieselbe Struktur erkannt wird.

Selbsttestaufgabe 1.3:

Welche Methoden, mit denen implizites Wissen in explizites Wissen umgewandelt werden kann, beschreibt das SECI-Modell?

Dokumente strukturieren

Bisher wurde das Erfassen der in einem Dokument enthaltenen Information von der Warte der menschlichen Lesenden aus betrachtet. Was ist aber, wenn Computerprogramme zur Informationsverarbeitung herangezogen werden sollen? Wollte man den menschlichen Prozess des Textverstehens nachbilden, der bei einer formatierten und für das menschliche Wahrnehmungssystem „Auge" realisierten Version des Dokuments startet, so müsste man Computerprogramme mit intelligenten Fähigkeiten ausstatten, die Zeichenerkennung, Sprachverstehen und Strukturerkennung einschließen. Von diesen Möglichkeiten ist man heute jedoch noch relativ weit entfernt, sie sind allerdings bereits Gegenstand intensiver Forschung.

Strukturebene des Dokumentes

Der Ansatz des Modells der strukturierten Dokumente sieht neben der ersten Ebene des rein textuellen Inhaltes folgende zwei weitere Ausgangspunkte für die computergestützte Informationsverarbeitung vor: zunächst eine explizite Repräsentation der Struktur des Dokumentes (zweite Ebene) im Sinne einer strukturellen Repräsentation. D. h., Autoren eines strukturierten Dokumentes editieren den eigentlichen Inhalt des Dokumentes und markieren die notwendigen Strukturelemente.

Formatebene des Dokumentes

Daraufhin folgt die dritte Ebene im Modell der strukturierten Dokumente, die sich auf die Formatierung der bereits strukturierten Inhalte bezieht. Das zu den Textelementen passende Format kann aus einer separaten, vom Dokument unabhängigen und bei Bedarf auswechselbaren Formatvorlage, auch Stylesheet genannt, berechnet werden.

Das Stylesheet enthält allgemeine Regeln, wie die einzelnen Strukturelemente zu formatieren sind. Durch eine Regel kann beispielsweise festgelegt werden, dass Überschriften in einer fetten Kursivschrift mit einem gewissen Abstand zum umgebenden Text formatiert werden sollen.

Ein sogenannter *Formatierer* bringt das mit Strukturmarkierungen versehene Quelldokument mit dem Stylesheet zusammen und formatiert das Dokument entsprechend der Vorgaben in dem Stylesheet.

Vorteile des Dokumentenmodells

Ein solches Vorgehen hat als entscheidenden Vorteil die Flexibilität, mit welcher die Formatierung der Texte immer wieder geändert werden kann. Die logische Strukturie-

rung unterstützt aber auch Anwendungen der Dokumentenverwaltung, sodass etwa die automatische Generierung von Katalogeinträgen und Verzeichnissen möglich wird oder das Wiedergewinnen, sprich das Wiederfinden und Zugreifen von Information nach bestimmten logischen Kriterien. Die eigentliche Informationsverarbeitung kann sich also direkt auf die logische Struktur stützen. Die Unabhängigkeit der Stylesheets von den Dokumenten ermöglicht, dass ein Dokument durch Kombination mit verschiedenen Stylesheets passend zum Anwendungszweck formatiert werden kann. Außerdem können verschiedene Dokumente durch Kombination mit ein- und demselben Stylesheet einheitlich formatiert werden. Auch lässt sich die Struktur beim Erstellen und Lesen von Dokumenten nutzen, etwa zur Navigation und zur Bereitstellung von Übersichten. Von diesen Vorteilen profitieren besonders die Anwendungen, die große Dokumentenbestände erstellen und verwalten müssen.

Eine umfangreiche Sammlung von Dokumenten mit Projektvorschlägen, die alle nach dem gleichen Muster logisch strukturiert und markiert sind, kann so z. B. systematisch nach anwendungsnahen Kriterien abgefragt werden. Auch etwa die im letzten halben Jahr neu hinzugekommenen Vorschläge können mit Titel und Datum ausgegeben werden. Eine weitere Anwendung wäre, automatisch nach allen Dokumenten mit Projekten zu suchen, die ein bestimmtes gemeinsames Ziel verfolgen. Auch für die Verwaltung des Dokumentenbestandes bietet das Modell der strukturierten Dokumente Vorteile. Die mit der Verwaltung befassten Personen brauchen für den gesamten Bestand nur einmal die zu verwendenden Strukturelemente festzulegen. Für jedes Präsentationsmedium ist nur ein einziges Stylesheet nötig, welches dann auf alle Dokumente des Bestandes anwendbar ist. Für das Beispiel der Projektvorschläge würden zwei Stylesheets ausreichen, eines für das Lesen am Bildschirm und eines für das Steuern eines Ausdrucks.

Das Modell der strukturierten Dokumente wurde in den achtziger Jahren entwickelt [AFQ89] [FHVV11] [MvD82]; seine Wurzeln gehen sogar bis in die sechziger Jahre zurück [E63]. Alle modernen Textverarbeitungssysteme unterstützen das Modell zumindest in Ansätzen.

Verfeinerungen des Dokumentenmodells

freie und vordefinierte Strukturlemente

Zum vollen Modell der strukturierten Dokumente gehört wesentlich dazu, dass die Namen der Strukturelemente frei wählbar sind. Wenn man beispielsweise ein Aufgabenblatt logisch strukturieren möchte, dann müssen die Strukturelemente wie Aufgabe, Abgabedatum, Punktezahl, Lösungshinweis oder Schwierigkeitsgrad angegeben werden können. Ist der Sprachvorrat, aus dem Strukturelemente gewählt werden können, fest vorgegeben und beschränkt, kann er nicht alle sich im Laufe der Zeit ergebenden Anwendungen abdecken. Dies verwässert allerdings die Vorteile der strukturierten Dokumente für die computergestützte Informationsverarbeitung, da semantische Information über die Rollen der Strukturelemente durch eventuelle Mehrfachbelegung verloren geht.

Eine weitere Überlegung zum Modell der strukturierten Dokumente ist, ob der zugrundeliegende Vorrat an Strukturelementen formal definiert werden soll. Eine Strukturvorgabe ist eine mit formalen Methoden festgelegte Definition des Vorrats an Strukturelementen, die auch Einschränkungen zur Verwendung der Elemente beinhalten kann. Eine Strukturvorgabe hat Vorteile. Beispielsweise können Dokumente automatisch auf ihre strukturelle Korrektheit überprüft werden oder Editierhilfen können solche Vorgaben auswerten und zur Unterstützung der Autorin oder des Autors einsetzen. Schließlich wird durch eine formale Strukturvorgabe auch das Vokabular einer Dokumentenanwendung dokumentiert.

Strukturdarstellungen

Zur Markierung der Strukturelemente gibt es viele verschiedene Konzepte. Bei den meisten Konzepten wird davon ausgegangen, dass die logische Strukturierung streng hierarchisch angelegt ist. Strukturelemente können also andere Strukturelemente vollständig enthalten, sich aber nicht mit ihnen überlappen. Die Strukturelemente, die zum Beispiel in der Beschreibung des Lehrgebiets Multimedia und Internetanwendungen in Abbildung unter Abschnitt 7.c eingeführt worden sind, gehorchen dem Gesetz der hierarchischen Schachtelung.

Häufig wird die hierarchische Struktur auch grafisch dargestellt. Hierzu können exemplarisch zunächst drei Varianten vorgestellt werden. Abbildungen zu allen drei Varianten befinden sich im Anhang.

Bei der ersten Variante werden ineinander geschachtelte Rechtecke oder Boxen, die mit dem Namen des jeweiligen Elementes gekennzeichnet sind, genutzt. Die zweite – dynamische – Variante, bei der mit Einrückungen und Ikonen gearbeitet wird, könnte aus Dateibrowsern bereits bekannt sein. Eine dritte – ebenfalls dynamische – Variante besteht darin, die logische Struktur als einen Baum darzustellen. Die grafischen Möglichkeiten stellen hohe Anforderungen an die Formatierungsmöglichkeiten des darstellenden und, ja auch möglichst noch Interaktionen zulassenden Systems.

Tags

Eine nicht so anspruchsvolle Möglichkeit besteht darin, die hierarchische Struktur mittels einer sogenannten Markierung (engl. *tag*) im Text zu versehen und damit zu serialisieren. Zu diesem Zweck wird jeder Textbereich, der Inhalt eines logischen Elementes ist, an seinem Anfang und an seinem Ende mit einer öffnenden und einer schließenden Markierung versehen, die den Namen des entsprechenden Strukturelementes enthält. Eine solche Markierung ist einfach ein sogenannter Klartext, der sich durch syntaktische Konventionen vom eigentlichen textuellen Inhalt abhebt. Der textuelle Inhalt ist ansonsten völlig unformatiert. Bei der XML-Konvention wird der Anfang einer Markierung durch einen Namen, der in spitzen Klammern steht, signalisiert (z. B. `<Titel>`). Soll das Strukturelement mit einer Markierung am Ende quasi abgeschlossen werden, also die strukturelle Markierung danach nicht mehr gelten, wird zusätzlich ein Schrägstrich in die Markierung eingesetzt (z. B. `</Titel>`). Der Oberbegriff zu Markierungen und eventuellen weiteren Möglichkeiten, nicht zum Inhalt zählende Zeichen in einen Text einzustreuen, ist der Begriff Auszeichnung (engl. *markup*). Eine grafi-

sche Verfeinerung dieser Strukturrepräsentation durch Auszeichnungen wird erreicht, wenn mit Einrückungen und mit typografischer Differenzierung von Text und Auszeichnung gearbeitet wird.

Da die Auszeichnungen mit im Text enthalten sind, wenn auch über syntaktische Konventionen vom eigentlichen Inhalt getrennt, spricht man auch von einer eingebetteten Auszeichnung (engl. *embedded markup*). Von einem prinzipiellen Standpunkt aus gesehen sind alle Darstellungsformen für strukturierte Dokumente äquivalent. Die Repräsentation der hierarchischen Struktur durch eingebettetes Markup ist technisch am einfachsten zu handhaben und ist deshalb derzeit noch die gängigste. Sie hat den Vorteil, dass Dokumente mit eingebettetem Markup mit jedem einfachen ASCII-Texteditor angesehen und bearbeitet werden können. Die Darstellungsweise ist allerdings unübersichtlich und nicht sehr eingängig. Die anderen Darstellungsformen stoßen an ihre Grenzen, wenn die Schachtelungstiefe sehr groß wird oder die Granularität der Struktur sehr fein. Probleme tauchen unter anderem schnell bei der logischen Beschreibung mathematischer Formeln auf.

eingebettete Auszeichnungen

Diskussion des Dokumentenmodells

Das Modell der strukturierten Dokumente wird eingesetzt, um Dokumente für die Computerverarbeitung vorzubereiten. Die zentrale Idee ist, die Kerndaten des Dokuments redundanzfrei in einer Art Reinform vorzuhalten, wie man es auch beim Design relationaler Faktendatenbanken mit Normalformen verfolgt. Faktendaten, die sich aus dieser Quellform berechnen lassen oder die variabel gehalten werden sollen, werden nicht in der Quellform mit codiert sondern bei Bedarf berechnet und gegebenenfalls als Sichten generiert. Verarbeitungsfunktionen können auf dieser Reinform der Daten aufsetzen und werden nicht belastet durch Idiosynkrasien im Datenbestand, die durch einzelne Anwendungen begründet sind.

Im Datenbankbereich dient diese Methode gleichzeitig der logischen Datenunabhängigkeit. Änderungen an der Struktur der Quelldaten können vor Endsystemen, die über Sichten mit der Datenbasis zu tun haben, verborgen bleiben. Das Potential der Datenunabhängigkeit ist durch das Modell der strukturierten Dokumente und seine Umsetzung mit XML auch im Bereich der Webdokumente gegeben. In diesem Bereich sind noch interessante Forschungsarbeiten zu erledigen. Auch eine Theorie der Normalformen, wie sie im Datenbankbereich etabliert ist, gibt es für den Bereich der strukturierten Dokumente noch nicht. Dennoch, gerade die Forschung in diesem Bereich und aktuelle Entwicklungen im Bereich der dokumenten- und XML-basierten Datenbanken (z. B. BaseX, MongoDB etc.) sowie die Integration spezieller Datentypen und Indizes für Informationssysteme verschiedenster Art und Methoden zur Verarbeitung in relationalen Datenbankmanagementsystemen (z. B. Oracle XML DB) versprechen einiges für die Zukunft.

Das Modell der strukturierten Dokumente dient somit dazu, Inhalt und Struktur eines Dokumentes, auf denen die automatische Verarbeitung basieren soll, zu repräsentieren. Dabei ist es wichtig, die Strukturelemente je nach Anwendungsfall frei definieren, und die entsprechenden Definitionen, auch formal festschreiben zu können. Auf dieser

Basis können Verarbeitungsfunktionen aufgesetzt und Sichten generiert werden, wobei die Basis je nach Anwendungsfall durch zusätzliche und separate Steuerungsdaten ergänzt werden kann.

Dokumentenmanagement

Historisch stammt das Modell aus dem Bereich der Dokumentenverarbeitung und Dokumentenverwaltung (Dokumentenmanagement, engl. *document management*). Ein wichtiger Anwendungsfall in diesem Bereich ist die Formatierung. Vorgaben für die Formatierung werden in separaten und auswechselbaren Formatierungsvorschriften (engl. *stylesheets*) vorgehalten und Formate aus beiden Beschreibungen zusammen berechnet. Über die Generierung formatierter Sichten hinaus sind Systeme zur Unterstützung des Dokumentenmanagements typischerweise in der Lage, Nummerierungen, Verzeichnisse und Querverweise aus der expliziten Repräsentation von Inhalt und Struktur zu erzeugen. Dies geschieht unter Hinzunahme von oft ebenfalls in Stylesheets enthaltenen Steuerungsdaten.

Informations- und Wissensmanagement

Das Modell unterstützt jedoch nicht nur Anwendungsfälle des Dokumentenmanagements im engeren Sinne, sondern trägt auch Anwendungen im weiteren Bereich des Informations- und Wissensmanagements Rechnung. Das Modell der strukturierten Dokumente dient somit als Werkzeug, um Texte zu gut zu verwaltenden Texten weiterzuentwickeln und die maschinelle Lesbarkeit und automatisierte Verarbeitung zu ermöglichen.

Zudem sollen Bereiche im Inhalt eines Dokuments mit zusätzlicher semantischer Information markiert und die Bedeutung oder Rolle des entsprechenden Textstücks somit durch eine Annotation repräsentiert werden. Das Modell platziert sich damit in einem Spektrum, das von einer rein grafischen Repräsentation eines Dokuments, etwa als Pixelmuster oder als PostScript-Dokument, bis zu einer expliziten Repräsentation der Semantik, etwa als semantisches Netz, reicht.

Was in dieser Lehrveranstaltung Dokumentenmodell genannt wird, heißt in der älteren Literatur auch manchmal Dokumentenarchitektur. Der Begriff der Architektur macht klarer deutlich, worum es in einer ersten Annäherung an Dokumente und ihrer Formate geht: relevante Aspekte von Dokumenten sollen identifizierbar und das Zusammenwirken transparent gemacht werden. Es werden im weiteren Verlauf der Lehrveranstaltung auch Überlegungen dazu angestellt werden, welche abstrakten Modelle den sich im Internet manifestierenden Formaten zugrunde liegen. Es steht dabei nicht immer für jede Ebene der Modellierung eine separate Begrifflichkeit – etwa wie hier „Architektur" – zur Verfügung. Bei dem noch folgenden Modell für die Zeichencodierung wird jedoch sehr schnell klar, warum in dieser Lehrveranstaltung der Begriff Modell eingeführt wird und wir die Thematik nicht unter dem Begriff Architektur der strukturierten Dokumente erläutern.

Umsetzungen des Dokumentenmodells

Anhand der meisten modernen Textverarbeitungssysteme wird das Modell der strukturierten Dokumente in bestimmter Form umgesetzt. Die beiden Systeme LaTeX und

MS-Word decken vom Typ her zusammen das Spektrum von Textsystemen ab, die in akademischen und technischen Dokumentierungen sowie im Büro zu finden sind. Die folgende Einführung der beiden Beispielsysteme soll dabei aufzeigen, in welcher Weise Verarbeitungssoftware für Dokumente heute das Modell der strukturierten Dokumente erfüllt.

Die Diskussion soll darüber hinaus auch dafür sensibilisieren, bei der Erstellung von neuen eigenen Dokumenten die angebotenen Mechanismen eines Dokumentensystems zur Unterstützung des Modells der strukturierten Dokumente möglichst weitgehend auszunutzen. Dies ist im Interesse eines effizienteren Dokumentenmanagements und einer flexibleren Weiterverwendung der daraus resultierenden Dokument- und Datensammlungen.

Umsetzung des Dokumentenmodells mit LaTeX

Das Beispiel der Beschreibung des Lehrgebiets Multimedia- und Internetanwendungen befindet sich im LaTeX-Format im Anhang unter 7.e.: Der textliche Inhalt ist gleich dem in der Anzeige unter 7.a und dem im XML-Format unter 7.d. Es ist auch eine gewisse Übereinstimmung bei den markierten Strukturen festzustellen. Die in der LaTeX-Version explizit markierten logischen Elemente sind z. B. die Abschnittüberschriften `\section{...} und \subsection{...}`, die Liste `\begin{itemize} ... \end{itemize}`, die Aufzählungspunkte `\item` und das Dokument als Ganzes `\begin{document} ... \end{document}`.

Die syntaktischen Formen des Markups in LaTeX sind vielfältiger als in XML, auch wenn es gewisse Ähnlichkeiten gibt. Eine direkte Entsprechung zu XMLs Konstrukt `<XXX>...</XXX>`, um ein logisches Element `XXX` zu markieren, ist `\begin{XXX} ... \end{XXX}`, aber es existieren auch die Formen `\XXX{...}` und sogar `\XXX ...` ohne explizite Markierung des Elementes.

Die Funktionsbedeutung ergibt sich aus der Anfangsmarkierung eines Elements zusammen mit der Metainformation, ob dieses nächste Element im Element XXX enthalten sein kann. Einen Sonderfall stellt die implizite Abgrenzung von Absätzen durch Leerzeilen dar.

Das zur Verfügung stehende Repertoire von logischen Elementen ist aus der zugrundeliegenden Dokumentenklasse `\documentclass{...}` und den hinzugeladenen Paketen `\usepackage[...]{...}` ersichtlich. Hier werden externe Dateien referenziert, die systemweit zur Verfügung stehen können. Es werden die Sprachelemente definiert, die im Dokument vorkommen.

Bei LaTeX ist also vorgesehen, dass die Autoren ihr eigenes Vokabular von Elementen definieren und dass eine Gruppe von Autoren/-innen ein gemeinsames Vokabular benutzt. Dieses wird dann nur einmal definiert und kann von verschiedenen Dokumenten aus referenziert werden. Das Vokabular selbst ist formal definiert und weitere syntaktische Einschränkungen sind implizit.

Mit LaTeX wird somit die erste Anforderung des Modells der strukturierten Dokumente erfüllt: Der Inhalt und die logische Struktur eines Dokumentes werden explizit repräsentiert. Die Strukturelemente können je nach Anwendungsfall frei definiert werden. Ansatzweise ist es auch möglich, die Strukturelemente formal festzulegen.

Wie sieht es mit der zweiten Anforderung aus, dass Vorgaben für die Formatierung in separaten und auswechselbaren Stylesheets vorzuhalten sind? Hier kommen wieder die Konstrukte `\documentclass{...}` und `\usepackage[...]{...}` ins Spiel, die schon bei der Definition der Strukturelemente herangezogen worden sind. Die hier referenzierten externen Dateien erfüllen demnach eine Doppelfunktion: Die Festlegung darauf, welche Strukturelemente im Dokument vorkommen können und die Vorgabe, wie die Strukturelemente formatiert werden sollen.

Die Forderung nach Stylesheets ist erfüllt. Im Falle von LaTeX sind diese extern und somit auswechselbar oder intern und an das Dokument gebunden.

In einer reinen Markupsprache werden nur die Markierungen bereitgestellt, die im Dokument verwendet werden. Bei LaTeX wird die Funktion von Markupsprache, Formatdefinition und Programmierung untrennbar aneinandergekoppelt. Dies muss nicht unbedingt problematisch sein – allerdings widerspricht es dem allgemeinen und wohlbegründeten Informatik-Prinzip, der Trennung von konzeptuellen Dimensionen. In der Praxis bedeutet das, dass eine Person, die ein Stylefile für LaTeX erstellen möchte, über ein ganzes Spektrum von Fähigkeiten und Fertigkeiten verfügen muss. Die gewünschte logische Struktur muss geplant und organisiert werden, das grafische Design muss entworfen und die so entwickelten Vorstellungen sollten dann auch durch Computerprogramme realisiert werden. Die Einstiegshürde für einen kreativen Gebrauch von LaTeX ist also hoch. Im Gegenzug ist LaTeX durch seine Programmierbarkeit ein mächtiges und in seiner Funktionalität bei Bedarf beträchtlich erweiterbares System.

LaTeX ist ursprünglich als Textsatzsystem entwickelt worden. Die Funktionen beschränken sich also im Wesentlichen auf den Bereich des Dokumentenmanagements. Da jedoch bei LaTeX die Technik des eingebetteten Markups zur Repräsentation von Strukturelementen zur Verfügung steht, können mithilfe von weiteren Programmen sowie entsprechenden Schnittstellen die LaTeX-Quelldokumente als Text eingelesen und automatisch weiterverarbeitet werden.

Die LaTeX-Software ist frei verfügbar. Die deutsche TeX Users Group DANTE e.V. stellt LaTeX für verschiedene Plattformen unter http://www.dante.de zur Verfügung.

Zusammenfassend lässt sich also feststellen, dass LaTeX das Modell der strukturierten Dokumente weitgehend erfüllt. Die Markupsprache besteht jedoch nicht aus einfach deklarativen Markierungen, sondern ist eine kommandoorientierte Textsprache und in wesentlichen Teilen sogar quasi eine Programmiersprache für diesen Zweck.

Umsetzung des Dokumentenmodells mit Microsoft Word

Das System Microsoft Word (ursprüngliche Version vor OpenOffice-XML-Standard), das nun näher betrachtet wird, steht LaTeX in seinen grundlegenden Eigenschaften fast diametral entgegen. Die Einfachheit der Benutzung als Bürosystem ist oberstes Ziel, für das Abstriche bei der Mächtigkeit, Flexibilität und Erweiterbarkeit in Kauf genommen werden.

Ein Word-Dokument wird in einer bereits formatierten Sicht editiert. Markup wird in der Grundeinstellung nicht angezeigt. Die zugeordnete Formatierung wird gleich umgesetzt und am Bildschirm angezeigt. Das sogenannte WhatyouSeeisWhatYouGet-Prinzip (WYSIWYG-Prinzip) ist somit so gut wie perfekt umgesetzt wie die Abbildung unter 7.a zeigt.

What you see ist what you get

Der Anhang enthält bei Abbildung im Abschnitt 7.c ein Bildschirmfoto des Beispiel-Dokumentes mit struktureller Unterteilung. Wer ein Microsoft Word-System installiert hat, kann vielleicht nachvollziehen, welche Strukturelemente im vorliegenden Beispiel verwendet wurden: Autorin, Titel, Hauptüberschrift, Nebenüberschrift, Datum, Version, Abschnittüberschrift, Absatz und Aufzählungspunkt.

Autoren/-innen können mit Word Zeichenketten oder ganzen Absätzen eine Formatierung zuweisen. Die unterschiedlichen Formatierungen werden als Satz von Formatier-Parametern mit den jeweiligen Werten unter einem individuellen Bezeichner abgespeichert.

Vom Modell der strukturierten Dokumente aus gesehen entsprechen die Namen der Formatvorlagen dem Vokabular an Strukturelementen, die Formatvorlage selbst entspricht einer Regel im Stylesheet und die Zuweisungen der Formatvorlagen an die Absätze und Zeichenketten im Dokument entsprechen der Markierung mit logischen Strukturelementen. Mit Word wird damit die explizite Repräsentation von Inhalt und logischer Struktur eines Dokumentes ermöglicht. Die Strukturelemente können je nach Anwendungsfall frei definiert werden, und ansatzweise ist es möglich, die Strukturelemente formal festzulegen.

Word und Stylesheets

Darüber hinaus ist es in Word möglich, in sogenannten Musterdokumenten (engl. *document templates*), die quasi als Stylesheets fungieren, Sammlungen von Formatvorlagen anzulegen, sie zu verwalten und sie Dokumenten zuzuordnen. Stylesheets in Word sind auf diese Weise auswechselbar.

Das Word-Dateiformat auf Basis von XML ist jetzt bereits seit einiger Zeit das annähernd standardkonforme Format und zugleich entsteht ein ZIP-Archiv. Wenn ein solches Archiv geöffnet oder extrahiert wird, findet man eine mehrstufige Hierarchie von Komponenten, die über entsprechende Auflistungen zusammengehalten werden. In Form sogenannter <Relationship>-Elemente werden die Beziehungen der Komponenten aufgezeigt und befinden sich im Ordner „rels". Einzelne Komponenten können ausgetauscht werden, ohne auf die Office-Anwendung zurückzugreifen. Der Kern des Do-

kuments ist die Datei „document.xml", die Textdaten sowie Format- und Zeichensatzeinstellungen enthält. Diese Datei ist der einzige erforderliche Teil. Falls Kopf- und Fußzeilen vorhanden sind, findet man diese in separaten Dateien: „header.xml" bzw. „footer.xml". Die Datei „fontTable.xml" enthält die Liste der Zeichensätze (engl. *fonts*); in der Datei „settings.xml" befinden sich die anwendungsspezifischen Einstellungen, und in der Datei „styles.xml" sind die verwendeten Formate. Bilder oder andere Mediendateien werden im Ordner „media" binär abgelegt. Mit jeder neuen Word-Version wird das Modell der strukturierten Dokumente somit besser umgesetzt. Die aktuelle Version ist Word 2021.

Selbsttestaufgabe 1.4:

Wie unterscheiden sich die im Lehrveranstaltungstext beschriebenen Konzepte zur grafischen Markierung der Strukturelemente eines Dokuments gegenüber der Markierung durch Auszeichnung?

Selbsttestaufgabe 1.5:

Beschreiben Sie die drei Ebenen des Modells der strukturierten Dokumente, die im Lehrveranstaltungstext genannt werden, in eigenen Worten.

5. Zeichencodierungen am Beispiel Unicode

glatter Text

Der Inhalt ist ein grundlegender Bestandteil eines strukturierten Dokuments. Die Repräsentation im Computer, also die codierte Datei, spielt somit auch eine wichtige Rolle. Es wird dabei davon ausgegangen, dass der Inhalt aus sogenanntem „glatten Text" besteht. Dies ist eine Folge von elementaren Texteinheiten wie Buchstaben, Ziffern, Interpunktionszeichen und anderen Zeichen. Diese Zeichen müssen letztendlich in Bits und Bytes codiert werden. Eine Zeichencodierung gibt dazu eine Vorschrift an.

Die bekannteste Zeichencodierung ist US-ASCII, standardisiert als ANSI X3.4 im Jahre 1968. Sie ist eine Variante von ISO 646-US und somit auch von ISO 646 aus dem Jahre 1972. Der Zeichensatz von US-ASCII in Abbildung 5.1 reicht lediglich aus, um lateinische und US-amerikanische Texte sowie Texte in einigen wenigen weiteren Sprachen zu codieren.

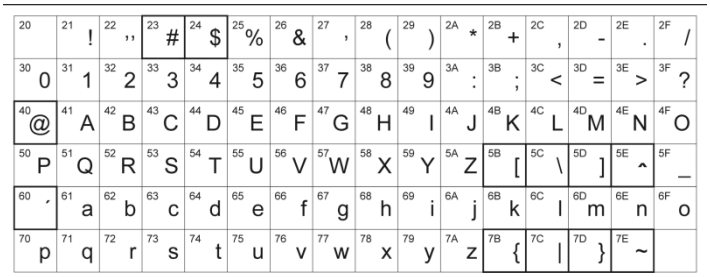


Abbildung 5.1 Der Zeichensatz ISO 646-US (US-ASCII)

Anfang der 1990er kam u. a. Unicode heraus. Das Ziel dieser Zeichencodierungen ist Universalität, also Texte aus sämtlichen Sprachen der Welt eindeutig zu codieren. Die aktuelle (März 2026) Version ist Unicode 17.0[2].

Unicode ist von zentraler Bedeutung für die Dokumentrepräsentation in den im Internet häufig verwendeten Auszeichnungssprachen HTML und XML.

Selbsttestaufgabe 1.6:

Was versteht man unter dem Begriff „glatter Text"? Wozu benötigt man in diesem Zusammenhang eine Zeichencodierung?

Im Folgenden wird Unicode genauer vorgestellt, und zwar im Rahmen eines abstrakten Codierungsmodells für Zeichen [DYIWFT02] [FSS82] [WD00]. Das Codierungsmodell stellt einen konzeptuellen Rahmen dar, innerhalb dessen man die Codierung von Zeichen in Bitmuster strukturieren und so besser verstehen kann. Die einzelnen Bestandteile des Codierungsmodells sind:

- ein abstrakter Zeichensatz,
- eine Codetabelle,
- ein Codierungsformat,
- ein Codierungsschema und
- eine Übertragungssyntax, die über sogenannten „glatten Text" hinausgeht.

generisches Zeichensatzmodell

Die einzelnen Bestandteile des Codierungsmodells entsprechen zunehmend konkreteren Repräsentationen von Zeichen, von einem abstrakten Begriff hin zu Bitmustern. Zwischen einer abstrakten Repräsentationsebene und der nächstliegenden konkreteren Repräsentationsebene besteht eine Abbildungsvorschrift im mathematischen Sinne. Um über mehrere Ebenen hinweg von einer Zeichenposition in einer Codetabelle direkt zu seiner Codierung in einem Codierungsschema zu gelangen, können die verschiedenen Abbildungsvorschriften konkateniert werden. Daraus entsteht eine sogenannte Zeichenkarte.

Abstrakter Zeichensatz

abstrakter Zeichensatz

Die obere abstrakte Ebene des Codierungsmodells ist der abstrakte Zeichensatz. Dieser besteht aus einer Menge sogenannter abstrakter Zeichen. Ein abstraktes Zeichen ist eine Informationseinheit, die zur Repräsentation, Organisation oder Kontrolle von Text dient. Es handelt sich hierbei also um Buchstaben, Ziffern, Interpunktionszeichen, Akzente, grafische Symbole, ideografische Zeichen, Leerzeichen, Tabulatoren, Zeilenweiterschaltung und -vorschub und

[2] https://www.unicode.org/versions/Unicode17.0.0/

Kontrollcodes für Auswahl-Markierungen. Der Zeichensatz ISO 64-US (US-ASCI)) beispielsweise besteht aus 33 Kontrollzeichen und 95 druckbaren Zeichen.

Ein abstrakter Zeichensatz beinhaltet jedoch noch keine Ordnung oder Nummerierung der im Zeichensatz enthaltenen Zeichen. Dieser Aspekt kommt in der nächsten Stufe des Codierungsmodells in Form einer sogenannten Codetabelle hinzu.

Abstrakte Zeichensätze wurden in der Regel festgelegt, um alle Texte einer bestimmten Sprache oder Sprachfamilie angemessen zu codieren. Der Anspruch, alle lebenden und alle historisch bedeutsamen Sprachen mit einem einzigen Zeichensatz codieren zu können, führte zu den (identischen) Zeichensätzen von Unicode und ISO/IEC 10646, dem sogenannten Universal Character Set, kurz UCS, genannt.

Zur visuellen Präsentation von Text dienen abstrakte grafische Einheiten, die Glyphen genannt werden. Glyphen entsprechen auf der Präsentationsebene den Zeichen auf der Codierungsebene. Die Abbildung 5.2 enthält grafische Ausprägungen der Glyphen für den Buchstaben A, sogenannte Glyphenbilder. Nach Design und anderen Charakteristika zusammengehörige Glyphenbilder bilden einen Font.

Abbildung 5.2 Typografie und Auswirkungen

Unicode formuliert zehn Designprinzipien für den Unicode-Standard. Zwei davon lassen sich bereits mit den Begriffen auf der Ebene des Zeichensatzes formulieren:

- Unicode codiert Zeichen, nicht Glyphen.
- Unicode codiert glatten Text, keine Formatinformation.

Glyphen spielen bei der Formatierung von Texten eine Rolle und sind auch bei der automatischen Texterkennung interessant. Für die computerisierte Weiterverarbeitung von Inhalten sind sie naturgemäß nicht besonders wichtig.

Codetabelle

Codetabelle

Auf der nächsten Stufe des Codierungsmodells steht die Codetabelle, die den abstrakten Zeichen im Zeichensatz eine Codeposition zuweist. Eine Codeposition ist immer eine natürliche Zahl, also größer oder gleich 0 (Für diese Lehrveranstaltung ist die Menge der natürlichen Zahlen so definiert, dass sie die 0 enthält: Es gilt demzufolge IN entspricht IN^{+0}).

Der Code-Raum einer Zeichen-Codierung ist immer ein Abschnitt der natürlichen Zahlen, der alle Codepositionen enthält. Der Code-Raum kann jedoch auch Zahlen enthalten, die keine zulässigen Codepositionen sind.

Codepositionen werden üblicherweise in Hexadezimalnotation angegeben, wodurch eine natürliche Beziehung zu Bitmustern hergestellt wird. Diese Beziehung kommt auf der nächsten Stufe des Codierungsmodells – dem Codierungsformat – zum Tragen.

Wie in Abbildung 5.1 gezeigt, umfasst der Code-Raum für US-ASCII 128 Positionen. In hexadezimaler Schreibweise geht der Code-Raum also von 0 bis 7F.

Der Code-Raum für Unicode enthält die 65.536_d Positionen von 0 bis $FFFF_h$ plus die $1.048.576_d$ Positionen von 10000_h bis $10FFFF_h$ und der für ISO/IEC 10646 die $2.147.483.648_d$ Positionen von 0 bis $7FFFFFFF_h$.
Der Unicode-Code-Raum enthält Lücken an 2.048_d *Surrogatpositionen* von $D800_h$ bis $DFFF_h$ und an den Positionen $FFFE_h$ sowie $FFFF_h$. Diese Positionen sind für spezielle Funktionen vorgesehen, und es werden ihnen auch künftig keine Unicode-Zeichen zugeordnet werden.6.400 Unicode-Codepositionen von $E000_h$ bis $F8FF_h$ – also knapp 10 Prozent des Code-Raums von 0 bis $FFFF_h$ – sind für sog. private Zwecke reserviert. Die Positionen sind etwa für Logos, für Icons oder für in speziellen Notationen benötigte Zeichen, etwa wie bei Musik und Tanz, außerhalb des Unicode-Zeichensatzes freigegeben.

Von den übrigen 57.086_d Unicode-Positionen im Bereich von 0 bis $FFFF_h$ sind 65_d mit Kontrollzeichen belegt, 49.194_d Positionen ein druckbares Zeichen zugeordnet und 7.827_d Positionen noch frei für Erweiterungen von Unicode. Im nächsten Abschnitt bei der Diskussion von Codierungsformaten wird erläutert, wie der Unicode-Code-Raum im Bereich von 0 bis $FFFF_h$ durch zwei Bytes und im Bereich von 10000h bis $10FFFF_h$ durch vier Bytes repräsentiert wird. Derzeit, d. h. bis zur Unicode-Version 3.0, enthält die Unicode-Codetabelle nur Werte im Bereich von 0 h bis $FFFF_h$.

Unicode beinhaltet für jedes in seiner Codetabelle codierte Zeichen die folgenden Daten:

- die Codeposition des Zeichens,
- ein typisches Glyphenbild für das Zeichen,
- einen Namen,
- semantische Information.

Von den zehn Designprinzipien, die Unicode anführt, lassen sich fünf auf der Ebene der Codetabelle erklären:

1 **Unifikation:** Unicode repräsentiert Zeichen, die in verschiedenen Alphabeten vorkommen, aber vom Aussehen her ähnlich sind, nur einmal.
2 **Konvertierbarkeit zwischen etablierten Standards:** Eine eineindeutige Abbildung aus einem etablierten Standard nach Unicode soll ermöglicht werden. Zeichenpositionen aus einem einzelnen international verbreiteten Zeichensatz, die nach dem

Designprinzip der Unifikation eigentlich miteinander identifiziert werden müssten, erhalten trotzdem getrennte Unicode-Codepositionen.

3 **Semantik:** Unicode definiert semantische Eigenschaften für Zeichen.

4 **Dynamische Komposition:** Jedes Basiszeichen kann mit beliebig vielen Kombinationszeichen gleichzeitig gepaart werden.

5 **Charakterisierung äquivalenter Codierungen:** Für die verschiedenen Codierungen eines Basiszeichens mit Kombinationszeichen oder eines primären und eines aus Kompatibilitätsgründen in den Zeichensatz aufgenommenen Zeichens kann eine normalisierte Codierung gefunden werden.

Codierungsformat

Mithilfe des Codierungsformats für eine Codetabelle werden die Bitrepräsentationen für die Codeposition festgelegt. Dazu gibt es eine Code-Einheit, die in der Regel aus acht oder sechzehn Bits besteht. Durch das Codierungsformat werden dann die Positionen eines Code-Raums in Sequenzen von Code-Einheiten und somit in Bitmuster abgebildet. Wird jede Codeposition einer Codetabelle auf die gleiche Anzahl von Code-Einheiten abgebildet, sprechen wir von einem Codierungsformat fester Länge; andernfalls hat das Codierungsformat variable Länge.

Ein kanonisches Codierungsformat fester Länge ergibt sich aus der Dualzahldarstellung von Codepositionen. Die Anzahl der jeweils benötigten Code-Einheiten ergibt sich aus der Größe des Code-Raum und der Zahl der Bits in einer Code-Einheit. In Tabelle 5.1 gibt es eine Übersicht über die verschiedenen Codierungsformate.

Codierung/Codierungsformat	**Code-Raum/Code-Einheit/Länge**
US-ASCII / kanonisch	0-7F / 1Byte / fest (1)
ISO 8859-X / kanonisch	0-FF / 1 Byte / fest (1)
SO / IEC 10646, Ebene 0 / UCS-2	0-FFFF / 1 Word / fest (1)
ISO / IEC 10646 / UCS-4	0-7FFFFFFF / 2 Word / fest (1)
Unicode 3.0 / UTF8	0-10FFFF / 1 Byte / variabel (1-4)
Unicode 3.0 / UTF16	0-10FFFF / 1 Word / variabel (1-2)

Tabelle 5.1 Codierungsformate für verschiedene Codierungen

Prinzipiell kann eine Codierung mehrere Codierungsformate definieren und das kanonische Codierungsformat muss nicht darunter sein. Unicode 3.0 beispielsweise definiert nur zwei Codierungsformate, nämlich UTF8 und UTF16, wobei UTF für Unicode Transfer Format steht. UTF8 hat eine Code-Einheit von 8 Bits Länge und repräsentiert Codepositionen mit einer bis vier Code-Einheiten. UTF16 hat eine Code-Einheit von 16 Bits Länge und repräsentiert Codepositionen mit einer bis zwei Code-Einheiten.

Ursprünglich war Unicode für einen Code-Raum von 0 bis $FFFF_h$ ausgelegt, sodass sechzehn Bits genügt hätten, um Codepositionen darzustellen. Tatsächlich hätte dieser Code-Raum ja auch mindestens bis Version 3.0 ausgereicht. Als jedoch klar wurde, dass zumindest für alle historisch interessanten Sprachen und Alphabete der Code-Raum nicht ausreichen würde und weil man mit ISO/IEC 10646 und seinem größeren Code-

Raum kompatibel bleiben wollte, wurde das Konzept der Surrogat-Codepositionen eingeführt und der Code-Raum von Unicode bis zur Position $10FFFF_h$ erweitert. Das Codierungsformat UTF16 benutzt jeweils ein Paar von Surrogatpositionen, um Positionen jenseits von $FFFF_h$ darzustellen.

UTF16 stellt wie die kanonische Codierung jede gültige Codeposition im Bereich von 0 bis $FFFF_h$ dar. Die ungültigen Codepositionen in dem sogenannten hohen Surrogatbereich von $D800_h$ bis $DBFF_h$ und in dem niedrigen Surrogatbereich von DC00 bis DFFF entsprechen den hohen und niedrigen Surrogat-Bitfolgen der Form $110110xxxxxxxxxx_b$ und $110111xxxxxxxxxx_b$, in denen jeweils zehn Bitpositionen für 2^{10} verschiedene Werte frei wählbar sind. Die Gegenrechnungen

$$DBFF_h + 1 - D800_h = DC00_h - D800_h = 400_h = 4x16^2_d = 2^{10}_d$$
$$DFFF_h + 1 - DC00_h = E000_h - DC00_h = 400_h = 2^{10}_d$$

bestätigen, dass in jedem der beiden Surrogatbereiche 2^{10}, also 1024_d ungültige Codepositionen liegen. Ein Paar von Surrogatpositionen, von denen die erste im hohen und die zweite im niedrigen Bereich liegt, ist somit eindeutig charakterisiert durch zwanzig Bits, die wiederum die 2^{20} Werte von 10000_h bis $10FFFF_h$ repräsentieren können:

$$10FFFF_h + 1 - 10000_h = 110000_h - 10000_h = 100000_h = 16^5_d = 2^{20}_d$$

Es ist deshalb stimmig, wenn bei UTF16 eine Codeposition P im Bereich von 10000_h bis $10FFFF_h$ durch die kanonische Bitdarstellung der beiden hohen und niedrigen Surrogatpositionen H und L repräsentiert, wobei sich H und L wie folgt berechnen:

Umgekehrt bestimmen eine hohe und eine niedrige Surrogatposition H und L eine Unicode-Position P im Bereich von 10000 bis $10\ FFFF_h$ wie folgt:

$$\mathbf{P = (H - D800_h) \bullet 400_h + (L - DC00_h) + 10000_h}$$

Das zweite Unicode-Codierungsformat, nämlich UTF8, hat die besondere Eigenschaft, die ersten 128 Codepositionen kanonisch mit einem Byte darzustellen. UTF8 ist damit US-ASCII-transparent.

Die Codierungseinheit von UTF8 ist 8 Bits lang. UTF8 stellt jede Position im Bereich von 0 bis $FFFF_h$ mit ein bis drei Codierungseinheiten, im Bereich von 0 bis $10FFFF_h$ mit ein bis vier Codierungseinheiten und im Bereich von 0 bis $7FFFFFFF_h$ mit ein bis sechs Codierungseinheiten dar.

Die UTF8-Darstellung einer Position mit einer einzigen Codierungseinheit hat die Form $0xxxxxxx_b$ mit einer führenden 0 und beliebigen Bits, die anstelle des Platzhalters x stehen. Die UTF8-Darstellung einer Position mit n Codierungseinheiten hat die Form $1^n0x^{7-n}{}_b$ gefolgt von n – 1 Codierungseinheiten der Form $10xxxxxx_b$. Bei drei Codierungseinheiten sind also genau sechzehn Bits frei wählbar, sodass mit drei Codierungseinheiten unter UTF8 Code-Positionen bis $FFFF_h$ darstellbar sind.

Die mit n Codierungseinheiten darstellbaren Positionsbereiche lassen sich aus der Tabelle 5.2 Grunddaten zu Codierungseinheiten entnehmen.

Einheiten	Darstellungsform	freie Bits	Maximum
1	0xxxxxxx	7	7F
2	110xxxxx 10xxxxxx	11	7FF
3	1110xxxx (10xxxxxx)2	16	FFFF
4	11110xxx (10xxxxxx)3	21	1FFFFF
5	111110xx (10xxxxxx)4	26	3FFFFFF
6	1111110x (10xxxxxx)5	31	7FFFFFFF

Tabelle 5.2 Grunddaten zu Codierungseinheiten

Für die Darstellung einer Codeposition unter UTF8 muss immer die kleinste Zahl von Codierungseinheiten gewählt werden. Eine Codeposition im Bereich von 800h bis $FFFF_h$ muss also mit drei Codierungseinheiten und darf nicht – obwohl das technisch möglich wäre – mit vier Codierungseinheiten dargestellt werden.

Die Umkehrfunktion von UTF8, die aus einer Folge von Code-Einheiten eine Codeposition berechnet, darf jedoch auch zu lange Darstellungen korrekt umrechnen. Die zwei Bytes $C1BF_h$ dürfen also unter UTF8 in die Position $7F_h$ zurückgerechnet werden, obwohl die UTF8-Darstellung der Position $7F_h$ das Byte $7F_h$ ist.

Das Unicode-Designprinzip der Effizienz lässt sich auf Ebene von Codierungsformaten erläutern. Für jedes der beiden Unicode-Codierungsformate UTF8 und UTF16 lässt sich in einem Strom von Positionsdarstellungen von jeder Codierungseinheit aus der Anfang der zugehörigen Positionsdarstellung mit beschränktem Backup finden. Im Falle von UTF16 muss höchstens um eine Codierungseinheit zurückgegangen werden, im Falle von UTF8 höchstens um drei Positionen.

Es sei noch angemerkt, dass UTF32 eine Zeichencodierung ist, welche im Gegensatz zu allen anderen UTF-Codierungen jedes Zeichen mit 4-Byte codiert und der UCS4-Codierung entspricht. Für Zeichen wie historische Schriftzeichen, alt- ägyptische Hieroglyphen oder seltene chinesische Schriftzeichen reichen 16 Bit oft nicht mehr aus. Hierfür wird jedes Zeichen mit 32 Bit codiert. UTF32 wurde zukunftsorientiert entwickelt. Da viele Bytes noch nicht belegt sind, ist es möglich, auch zukünftige Symbole, Zeichen, oder Sprachen mit zu integrieren. Dadurch, dass UTF32 keine variablen Bytelängen besitzt, ergibt sich der Vorteil, dass jedes Byte eindeutig zugeordnet ist und nicht an einer variablen Stelle sitzen kann.

Somit kann der Benutzende sehr einfach auf ein bestimmtes Zeichen im Speicherbereich zugreifen. Des Weiteren lässt sich durch die feste Speichervorgabe sehr einfach eine Textlänge ermitteln, indem der Gesamtspeicher lediglich durch vier Byte dividiert wird. Es sollte allerdings bei der Verwendung von UTF32 der wesentlich (!) höhere Speicherplatz-Bedarf bedacht werden.

In der Version ISO/IEC 10646-3:2003 werden die gleichen Formate UTF8, UTF16 und UTF32 beschrieben wie in Unicode 4.0. Seit Revision 2011 sind die Standards hinsichtlich der Codierungen deckungsgleich.

Codierungsschema

Damit Daten über Netzwerke zuverlässig ausgetauscht werden können, müssen sie in eine lineare Folge von Bytes gebracht werden. Dieser Vorgang wird auch Serialisierung genannt. Mithilfe eines Codierungsformats wird der Text als Folge von Codierungseinheiten dargestellt. Ein Codierungsschema wird nun dazu genutzt zu einem Codierungsformat zusätzlich festzulegen, wie die Codierungseinheiten in Bytefolgen zu serialisieren sind.

Serialisierung von Zeichencodierungen

Ist die Codierungseinheit eines Codierungsformats selbst acht Bits lang, so ist nichts mehr festzulegen und das Codierungsschema ist mit dem Codierungsformat identisch. So kann also UTF8 sowohl als Codierungsformat als auch als Codierungsschema bezeichnet werden.

Ist die Codierungseinheit ein Word, wie bei UTF16, so gibt es zwei Möglichkeiten der Serialisierung: Big Endian (das höherwertige Bit kommt zuerst) und Little Endian. Dementsprechend gibt es zu UTF16 zwei Codierungsschemata: UTF16-BE und UTF16-LE. Analoges gilt für UCS2, UCS4 und UTF32.

Codierungsschema

Mit dem sogenannten Byte-Reihenfolge-Cordierungszeichen (engl. *byte order mark*, *BOM*) an Position $FEFF_h$ kann bei einem UTF16-repräsentierten Datenstrom signalisiert werden, welche der beiden Serialisierungen, höherwertiges Byte zuerst (engl. *big endian*) oder niederwertiges Byte zuerst (engl. *little endian*), vorliegt. Da $FFFE_h$ keine zulässige Codeposition bei Unicode ist, lässt sich schließen, dass das Codierungsschema UTF16-LE vorliegt. Dies funktioniert jedoch nur dann, wenn als die ersten zwei Bytes FF_h und FE_h gelesen werden. Ein BOM am Anfang eines UTF16-codierten Datenstroms ist nicht Bestandteil der Daten und wird – abgesehen von seiner Signalwirkung für die Serialisierung – ignoriert. Tritt die Codeposition $FEFF_h$ dagegen an anderer Stelle im Datenstrom auf, so wird sie als das entsprechende Unicode-Zeichen Zero Width No-Break Space interpretiert, das dieser Position zugewiesen ist. Bei UTF-32 stehen davor oder dahinter noch zwei Nullbytes, die zur Erkennung der Byte-Reihenfolge dienen (UTF32-BE 00 00 FE FF, UTF32-LE FF FE 00 00).

BOM

Generell darf ein Unicode-Zeichenstrom am Anfang mit dem zusätzlichen BOM versehen werden. Anwendungen, die über keine Metainformation über die Natur des Datenstroms verfügen, erhalten so das Signal, dass es sich im Folgenden um Unicode-Daten, die in einem bestimmten Codierungsschema vorliegen, handelt. Das erste BOM eines Zeichenstroms bildet also keinen Teil der eigentlichen Daten.

Selbsttestaufgabe 1.7:

Welches sind die Bestandteile des im Lehrveranstaltungstext beschriebenen abstrakten Codierungsmodells? Erläutern Sie diese kurz in eigenen Worten.

6. Zusammenfassung und Ausblick

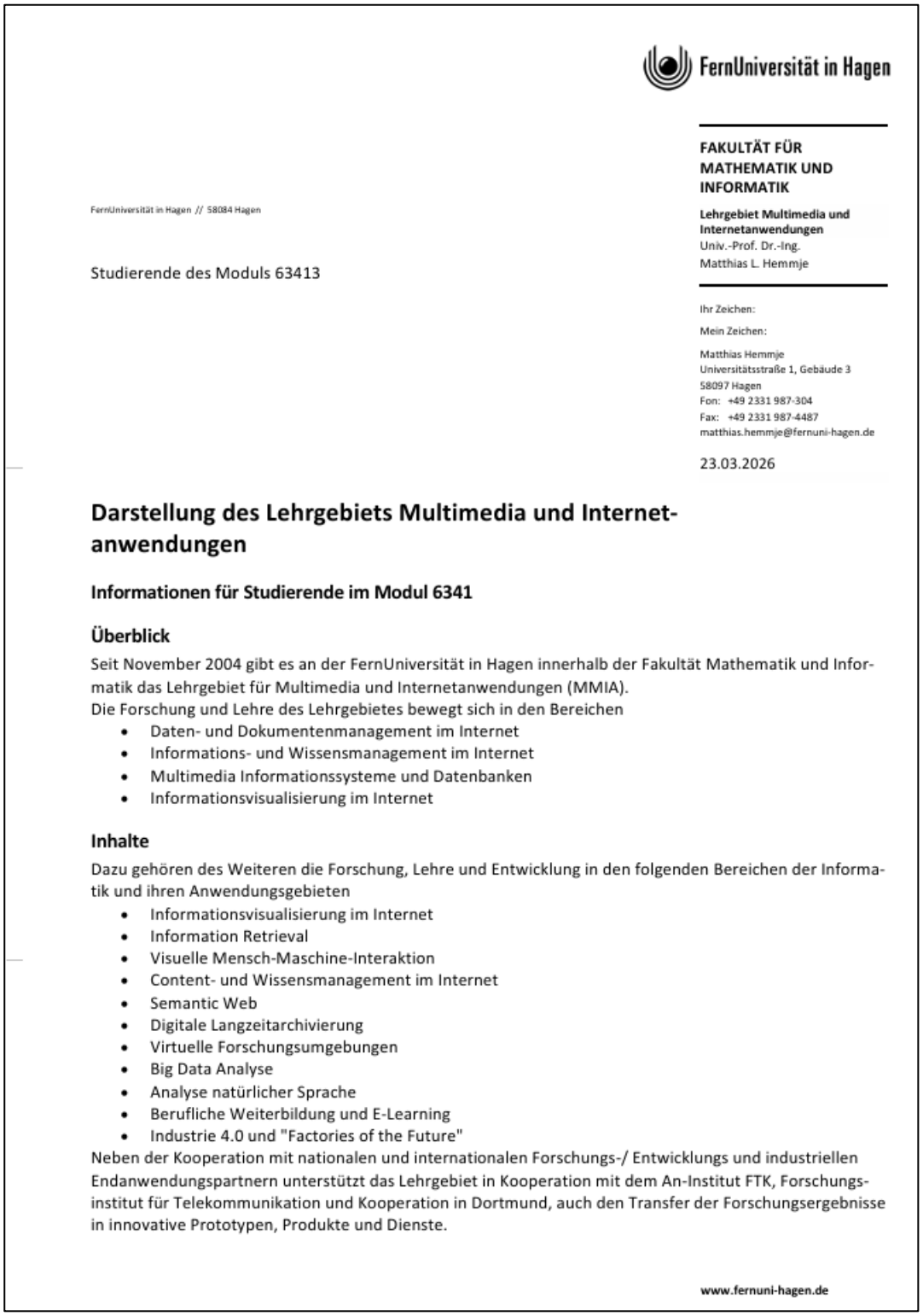
Zwei wichtige Faktoren machen digitale Dokumente zu modernen Informationsträgern, die vielfach verwendbar und der automatischen Verarbeitung zugänglich sind.

Die beiden Faktoren sind das Modell der strukturierten Dokumente, dem solche digitalen Dokumente entsprechen sollen, und die Standardisierung der Textcodierung, die Textinhalte austauschbar macht.

Beide Faktoren wurden in dieser Lektion besprochen. Als dritter Faktor steht die Standardisierung der Strukturinformation noch aus; damit werden die kommenden Lektionen im Zusammenhang mit XML und den verwandten Sprachfamilien zu tun haben.

7. Anhang: Formate am Beispiel

a. Fertig formatiertes Dokument



FernUniversität in Hagen

FAKULTÄT FÜR MATHEMATIK UND INFORMATIK

Lehrgebiet Multimedia und Internetanwendungen
Univ.-Prof. Dr.-Ing.
Matthias L. Hemmje

FernUniversität in Hagen // 58084 Hagen

Studierende des Moduls 63413

Ihr Zeichen:

Mein Zeichen:

Matthias Hemmje
Universitätsstraße 1, Gebäude 3
58097 Hagen
Fon: +49 2331 987-304
Fax: +49 2331 987-4487
matthias.hemmje@fernuni-hagen.de

23.03.2026

Darstellung des Lehrgebiets Multimedia und Internet-anwendungen

Informationen für Studierende im Modul 6341

Überblick

Seit November 2004 gibt es an der FernUniversität in Hagen innerhalb der Fakultät Mathematik und Informatik das Lehrgebiet für Multimedia und Internetanwendungen (MMIA).
Die Forschung und Lehre des Lehrgebietes bewegt sich in den Bereichen

- Daten- und Dokumentenmanagement im Internet
- Informations- und Wissensmanagement im Internet
- Multimedia Informationssysteme und Datenbanken
- Informationsvisualisierung im Internet

Inhalte

Dazu gehören des Weiteren die Forschung, Lehre und Entwicklung in den folgenden Bereichen der Informatik und ihren Anwendungsgebieten

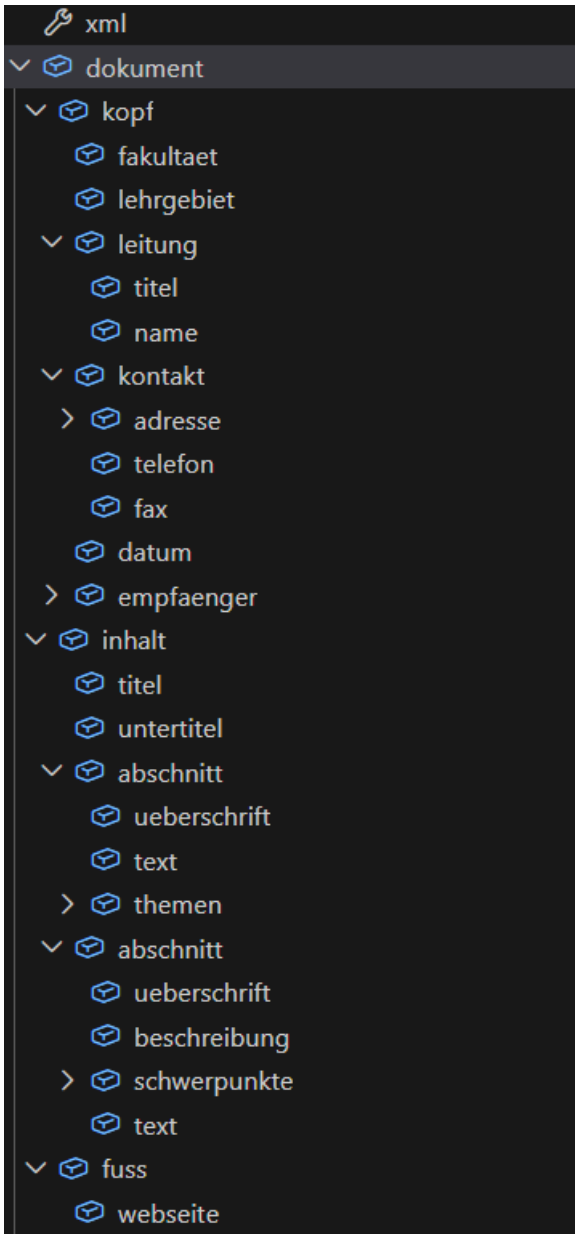
- Informationsvisualisierung im Internet
- Information Retrieval
- Visuelle Mensch-Maschine-Interaktion
- Content- und Wissensmanagement im Internet
- Semantic Web
- Digitale Langzeitarchivierung
- Virtuelle Forschungsumgebungen
- Big Data Analyse
- Analyse natürlicher Sprache
- Berufliche Weiterbildung und E-Learning
- Industrie 4.0 und "Factories of the Future"

Neben der Kooperation mit nationalen und internationalen Forschungs-/ Entwicklungs und industriellen Endanwendungspartnern unterstützt das Lehrgebiet in Kooperation mit dem An-Institut FTK, Forschungsinstitut für Telekommunikation und Kooperation in Dortmund, auch den Transfer der Forschungsergebnisse in innovative Prototypen, Produkte und Dienste.

b. Unformatiertes Dokument mit reinem Text

FAKULTÄT FÜR MATHEMATIK UND INFORMATIK FernUniversität in Hagen // 58084 Hagen Studierende des Moduls 63413
Betreff: Darstellung des Lehrgebiets Multimedia und Internetanwendungen Überblick Lehrgebiet Multimedia und Internetanwendungen Univ.-Prof. Dr.-Ing. Matthias L. Hemmje Ihr Zeichen: Mein Zeichen: Matthias Hemmje Universitätsstraße 1, Gebäude 3 58097 Hagen Fon: +49 2331 987-304 Fax: +49 2331 987-4487 matthias.hemmje@fernuni-hagen.de 23.03.2026 Seit November 2004 gibt es an der FernUniversität in Hagen innerhalb der Fakultät Mathematik und Informatik das Lehrgebiet für Multimedia und Internetanwendungen (MMIA). Die Forschung und Lehre des Lehrgebietes bewegt sich in den Bereichen Daten- und Dokumentenmanagement im Internet Informations- und Wissensmanagement im Internet Multimedia Informationssysteme und Datenbanken Informationsvisualisierung im Internet Inhalte Dazu gehören des Weiteren die Forschung, Lehre und Entwicklung in den folgenden Bereichen der Informatik und ihren Anwendungsgebieten Informationsvisualisierung im Internet Information Retrieval Visuelle Mensch-Maschine-Interaktion Content- und Wissensmanagement im Internet Semantic Web Digitale Langzeitarchivierung Virtuelle Forschungsumgebungen Big Data Analyse Analyse natürlicher Sprache Berufliche Weiterbildung und E-Learning Industrie 4.0 und "Factories of the Future" Neben der Kooperation mit nationalen und internationalen Forschungs-/ Entwicklungs und industriellen Endanwendungspartnern unterstützt das Lehrgebiet in Kooperation mit dem An-Institut FTK, Forschungsinstitut für Telekommunikation und Kooperation in Dortmund, auch den Transfer der Forschungsergebnisse in innovative Prototypen, Produkte und Dienste.

c. Dokumentdarstellung mit optischer Gliederung



d. Strukturcodierung mithilfe von eingebetteten Markierungen

```
<?xml version="1.0" encoding="UTF-8"?>
<dokument typ="geschaeftsbogen">

  <kopf>
    <fakultaet>Fakultät für Mathematik und Informatik</fakultaet>
    <lehrgebiet>Multimedia und Internetanwendungen</lehrgebiet>
    <leitung>
      <titel>Univ.-Prof. Dr.-Ing.</titel>
      <name>Matthias L. Hemmje</name>
    </leitung>
    <kontakt>
      <adresse>
        <strasse>Universitätsstraße 1</strasse>
```

```
      <gebaeude>Gebäude 3</gebaeude>
      <plz>58097</plz>
      <ort>Hagen</ort>
    </adresse>
    <telefon>+49 2331 987-304</telefon>
    <fax>+49 2331 987-4487</fax>
    matthias.hemmje@fernuni-hagen.de
  </kontakt>
  <datum>2026-03-23</datum>
  <empfaenger>
    <zielgruppe>Studierende des Moduls 63413</zielgruppe>
    <institution>FernUniversität in Hagen</institution>
    <plz>58084</plz>
    <ort>Hagen</ort>
  </empfaenger>
</kopf>

<inhalt>
  <titel>Darstellung des Lehrgebiets Multimedia und Internetanwendungen</titel>
  <untertitel>Informationen für Studierende im Modul 6341</untertitel>

  <abschnitt id="ueberblick">
    <ueberschrift>Überblick</ueberschrift>
    <text>
      Seit November 2004 gibt es an der FernUniversität in Hagen innerhalb der
      Fakultät Mathematik und Informatik das Lehrgebiet für Multimedia und
      Internetanwendungen (MMIA).
    </text>
    <themen>
      <thema>Daten- und Dokumentenmanagement im Internet</thema>
      <thema>Informations- und Wissensmanagement im Internet</thema>
      <thema>Multimedia Informationssysteme und Datenbanken</thema>
      <thema>Informationsvisualisierung im Internet</thema>
    </themen>
  </abschnitt>

  <abschnitt id="inhalte">
    <ueberschrift>Inhalte</ueberschrift>
    <beschreibung>
      Forschung, Lehre und Entwicklung in den folgenden Bereichen der Informatik
      und ihren Anwendungsgebieten:
    </beschreibung>
    <schwerpunkte>
      <schwerpunkt>Informationsvisualisierung im Internet</schwerpunkt>
      <schwerpunkt>Information Retrieval</schwerpunkt>
      <schwerpunkt>Visuelle Mensch-Maschine-Interaktion</schwerpunkt>
      <schwerpunkt>Content- und Wissensmanagement im Internet</schwerpunkt>
      <schwerpunkt>Semantic Web</schwerpunkt>
      <schwerpunkt>Digitale Langzeitarchivierung</schwerpunkt>
      <schwerpunkt>Virtuelle Forschungsumgebungen</schwerpunkt>
      <schwerpunkt>Big Data Analyse</schwerpunkt>
```

```
        <schwerpunkt>Analyse natürlicher Sprache</schwerpunkt>
        <schwerpunkt>Berufliche Weiterbildung und E-Learning</schwerpunkt>
        <schwerpunkt>Industrie 4.0 und Factories of the Future</schwerpunkt>
      </schwerpunkte>
      <text>
        Kooperation mit nationalen und internationalen Forschungs-,
        Entwicklungs- und Industriepartnern sowie mit dem An-Institut FTK
        (Forschungsinstitut für Telekommunikation und Kooperation in Dortmund)
        zum Transfer von Forschungsergebnissen in Prototypen, Produkte und Dienste.
      </text>
    </abschnitt>
  </inhalt>

  <fuss>
    <webseite>www.fernuni-hagen.de</webseite>
  </fuss>

</dokument>
```

e. Strukturcodierung mit LaTeX

```
\documentclass[a4paper,12pt]{article}

\usepackage[ngerman]{babel}
\usepackage[T1]{fontenc}
\usepackage[utf8]{inputenc}
\usepackage{geometry}
\geometry{margin=2.5cm}

\begin{document}

%-----------------------------------------------
% Kopfbereich
%-----------------------------------------------
\begin{center}
\textbf{Fakultät für Mathematik und Informatik}\\
Lehrgebiet Multimedia und Internetanwendungen\\[0.5em]

Univ.-Prof.\ Dr.-Ing.\ Matthias L. Hemmje\\[1em]
\end{center}

\noindent
Universitätsstraße 1, Gebäude 3\\
58097 Hagen\\
Fon: +49 2331 987-304\\
Fax: +49 2331 987-4487\\
E-Mail: \texttt{matthias.hemmje@fernuni-hagen.de}\\[1em]

\noindent
23.\,03.\,2026\\[1em]
```

```
\noindent
\textbf{Studierende des Moduls 63413}\\
FernUniversität in Hagen\\
58084 Hagen

\vspace{2em}

%-----------------------------------------------
% Inhalt
%-----------------------------------------------
\section*{Darstellung des Lehrgebiets Multimedia und Internetanwendungen}
\subsection*{Informationen für Studierende im Modul 6341}

\subsection*{Überblick}
Seit November 2004 gibt es an der FernUniversität in Hagen innerhalb der
Fakultät Mathematik und Informatik das Lehrgebiet für Multimedia und
Internetanwendungen (MMIA).

Die Forschung und Lehre des Lehrgebiets bewegt sich in den Bereichen:
\begin{itemize}
  \item Daten- und Dokumentenmanagement im Internet
  \item Informations- und Wissensmanagement im Internet
  \item Multimedia Informationssysteme und Datenbanken
  \item Informationsvisualisierung im Internet
\end{itemize}

\subsection*{Inhalte}
Dazu gehören des Weiteren die Forschung, Lehre und Entwicklung in den
folgenden Bereichen der Informatik und ihren Anwendungsgebieten:
\begin{itemize}
  \item Informationsvisualisierung im Internet
  \item Information Retrieval
  \item Visuelle Mensch-Maschine-Interaktion
  \item Content- und Wissensmanagement im Internet
  \item Semantic Web
  \item Digitale Langzeitarchivierung
  \item Virtuelle Forschungsumgebungen
  \item Big Data Analyse
  \item Analyse natürlicher Sprache
  \item Berufliche Weiterbildung und E-Learning
  \item Industrie 4.0 und Factories of the Future
\end{itemize}

\subsection*{Text}
Neben der Kooperation mit nationalen und internationalen Forschungs-,
Entwicklungs- und industriellen Endanwendungspartnern unterstützt das
Lehrgebiet in Kooperation mit dem An-Institut FTK (Forschungsinstitut für
Telekommunikation und Kooperation in Dortmund) den Transfer der
Forschungsergebnisse in innovative Prototypen, Produkte und Dienste.
```

```
\vfill
\begin{center}
\texttt{www.fernuni-hagen.de}
\end{center}

\end{document}
```

8. Literaturverzeichnis

[AFQ89] J. André, R. Furuta, and V. Quint. Structured Documents. The Cambridge Series on Electronic Publishing. Cambridge University Press, Cambridge, 1989.

[DYIWFT02] M. J. Dürst, F. Yergeau, R. Ishida, M. Wolf, A. Freytag, and T. Texin. Character model for the World Wide Web 1.0. www.w3.org/TR/charmod, February 2002. W3C Working Draft.

[E63] D. Engelbart. A conceptual framework for the augmentation of man's intellect. In P. W. Howerton and D. C. Weeks, editors, Vistas in Information Handling, volume 1, pages 1–29, Washington, DC, 1963. Spartan Books.

[FHVV11] T. Feuerstack, A. Hartmann, B. Vogeler, and J. Vieler. Einmal um die Erde und zurück: Unterwegs im Internet. Broschüre A/003/0811, Universitätsrechenzentrum FernUniversität Hagen, August 2011.

[FSS82] R. Furuta, J. Scofield, and A. Shaw. Document formatting: Survey, concepts, and issues. Computing Surveys, 14(3):417–172, September 1982

[MvD82] N. Meyrowitz and A. van Dam. Interactive editing systems (Parts I and II). Computing Surveys, 14(3):321–415, September 1982.

[WD00] K. Whistler and M. Davis. Character encoding model. Unicode Technical Report 17-3.1, Unicode, Inc., August 2000.

[Kuh89] Rainer Kuhlen. Pragmatischer Mehrwert von Information. Sprachspiele mit informationswissenschaftlichen Grundbegriffen. Universität Konstanz / Informationswissenschaft: Bericht;89,1. Konstanz: Universität Konstanz, 1989. URL: http://nbn-resolving.de/urn:nbn:de:bsz:352-247844.

[Wil95] Patrick Wilson. "Unused relevant information in research and development". In: Journal of the American society for Information Science 46.1 (1995), pp. 45–51. ISSN: 0002-8231

[NT95] I. Nonaka and H. Takeuchi, The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation, Oxford University Press, 1995.

[Vu15] B. Vu, "Realizing an Applied Gaming Ecosystem - Extending an Education Portal Suite towards an Ecosystem Portal," TU Darmstadt, 2015.

[Bat02] Marcia J. Bates. "Toward an integrated model of information seeking and searching". In: The New Review of Information Behaviour Research 3.1 (2002), pp. 1–15.

[Bat10] Marcia J. Bates. "Information behavior". In: Encyclopedia of library and information sciences 3 (2010). Publisher: CRC press New York, pp. 2381–2391.

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. Introduction to in-

formation retrieval. New York: Cambridge University Press, 2008. ISBN: 978-0-521-86571-5.

[BOB82] BELKIN, N.J., ODDY, R.N. and BROOKS, H.M. (1982), "ASK FOR INFORMATION RETRIEVAL: PART I. BACKGROUND AND THEORY", Journal of Documentation, Vol. 38 No. 2, pp. 61-71. https://doi.org/10.1108/eb026722

9. Abbildungsverzeichnis

Abbildung 4.1 Modi der Informationssuche [Bato2] 10
Abbildung 4.2 Das SECI Model [NT95] 12
Abbildung 4.3 Webseitengestaltung von 1995 13
Abbildung 5.1 Der Zeichensatz ISO 646-US (US-ASCII) 24
Abbildung 5.2 Typografie und Auswirkungen 26

10. Tabellenverzeichnis

Tabelle 5.1 Codierungsformate für verschiedene Codierungen 28
Tabelle 5.2 Grunddaten zu Codierungseinheiten 30

11. Lösungen der Selbsttestaufgaben

Selbsttestaufgabe 1.1

Erläutern Sie, wie von einer Suchmaschine Kuhlens „Primacy of Information Work" bei einer Suche nach dem Stichwort „NASA Perseverance" umgesetzt wird. Die Suche wird von einem Vater ausgeführt, um Fragen seiner Tochter zu der Mission zu beantworten.

- Die Suchanfrage wird durch ein Informationsbedürfnis auf Seite der Nutzenden ausgelöst, da diese die benötigte Information nicht haben, um das Problem (Nachfrage der Tochter) zu lösen, wie von Kuhlen skizziert („needed by but not needed by but not available to a specific person in a concrete situation to solve a problem").
- Das dafür benötigte Wissen liegt in einer Vielzahl von Quellen (z. B. NASA Homepage, wissenschaftliche Portale, Nachrichtenseiten) prinzipiell vor, steht den Nutzenden aber nicht zur Verfügung.
- Durch die Nutzung einer Suchmaschine und Zugriff auf die Ergebnisse erfolgt die Transformation des Wissens in Information für den Vater, die dieser wiederum nutzen kann, um seiner Tochter zu berichten.
- Damit findet auf diese Weise die Transformation von Wissen in Aktion (=Information) statt, was das Primacy of Information Work nach Kuhlen darstellt.

Selbsttestaufgabe 1.2

Überlegen Sie, ob auf die folgenden Dokumente eher der ursprüngliche, der neuere oder der moderne Dokumentenbegriff passt oder ob es sich vielleicht gar nicht um Dokumente handelt.

- Personalausweis
- Geldschein
- Gemälde
- Veranstaltungsplakat
- Privatbrief
- E-Mail-Nachricht
- Memo an die Geschäftsleitung
- Gedichtband
- Roman
- Bestellung
- Produktbeschreibung
- Fragebogen
- Nachricht in einem Protokoll zum digitalen Zahlungsverkehr (Electronic Banking)

Bei Gemälden handelt es sich wegen des fehlenden sprachlichen Ausdrucks nicht um Dokumente.

Personalausweise und Geldscheine sind eindeutig Dokumente ursprünglicher Art, für die das Trägermedium ebenso essenziell ist wie der sprachliche Informationsgehalt.

Veranstaltungsplakate, Privatbriefe, Gedichtbände und Romane sind wohl in der Regel Dokumente neuerer Art, in denen neben dem Informationsgehalt auch die Formatierung eine Rolle spielt.

Limitierte Auflagen und handschriftliche Briefe mit Unterschrift könnten wir jedoch genauso gut als Dokumente ursprünglicher Art einordnen.

Memos, Bestellungen, Produktbeschreibungen und Fragebögen sehe ich je nach Kontext als Dokumente neuerer oder moderner Art an. Ein wichtiges Entscheidungskriterium ist, ob diese Dokumente digital oder auf Papier vorliegen.

E-Mail-Nachrichten oder Nachrichten in einem Protokoll zum digitalen Zahlungsverkehr sind Nachrichten moderner Art, solange man den Protokollnachrichten den sprachlichen Ausdruck zugesteht.

Selbsttestaufgabe 1.3

Welche Methoden, mit denen implizites Wissen in explizites Wissen umgewandelt werden kann, beschreibt das SECI-Modell?

1. **Sozialisation** (implizit zu implizit): Menschen kommunizieren direkt mit Menschen oder wenden den Wissensaustausch durch Erfahrungsaustausch an. Heutzutage werden viele fortschrittliche Technologien eingeführt, aber das Sozialisieren bleibt wichtig und kann nicht ersetzt werden.

2. **Externalisierung** (implizit bis explizit): Zusammen mit der Erfindung des Schreibens kann Wissen vom Gehirn auf andere Medien wie z. B. Bücher, Videos usw. übertragen werden. Dieser Vorgang wird als Externalisierung bezeichnet.

3. **Kombination** (explizit zu explizit): Dieser Prozess besteht darin, das Wissen zu organisieren und Teile davon in neuen Formen zu kombinieren, damit es sinnvoller ist oder neue Erkenntnisse erklärt.

4. **Internalisierung** (explizit zu implizit): Personen können neues Wissen erwerben, indem sie explizites Wissen wahrnehmen. Einmal verinnerlicht, kann neues Wissen wiederverwendet, dokumentiert oder an andere Personen weitergegeben werden.

Selbsttestaufgabe 1.4

Wie unterscheiden sich die im Lehrveranstaltungstext beschriebenen Konzepte zur grafischen Markierung der Strukturelemente eines Dokuments gegenüber der Markierung durch Auszeichnung?

Übereinstimmend wird in den Konzepten davon ausgegangen, dass die logische Strukturierung hierarchisch angelegt ist. Bei grafischen Konzepten wird die hierarchische Struktur grafisch dargestellt. Es gibt drei Varianten:

- Bei der ersten Variante werden ineinander geschachtelte Rechtecke oder Boxen, die mit dem Namen des jeweiligen Elementes gekennzeichnet sind, genutzt.
- Die zweite Variante verwendet zur Darstellung der Struktur Einrückungen und Icons.
- Die dritte Variante besteht darin, die logische Struktur als einen Baum darzustellen.

Konzepte, die auf Markup basieren, bestehen darin, die hierarchische Struktur mittels einer sogenannten Markierung (engl. *tag*) im Text zu versehen: jeder Textbereich, der Inhalt eines logischen Elementes ist, wird an seinem Anfang und an seinem Ende mit einer öffnenden und einer schließenden Markierung versehen, die den Namen des entsprechenden Strukturelementes enthält.

Der Oberbegriff zu Markierungen und eventuellen weiteren Möglichkeiten, nicht zum Inhalt zählende Zeichen in einen Text einzustreuen, ist der Begriff Auszeichnung (engl. *markup*).

Selbsttestaufgabe 1.5

Beschreiben Sie die drei Ebenen des Modells der strukturierten Dokumente, die im Lehrveranstaltungstext genannt werden, in eigenen Worten.

Die erste Ebene des Modells der strukturierten Dokumente bezieht sich auf den Inhalt eines Dokuments. Gemeint sind damit die rein textuellen Inhalte in Form von Buchstaben und Zeichen, die beim Verarbeiten / Lesen zu Wörtern zusammengesetzt und mit Bedeutung versehen werden. (Inhaltsebene)

Die zweite Ebene (Strukturebene) bezeichnet die explizite Repräsentation der Struktur des Dokuments, die über das Einfügen von z. B. Markup realisiert wird.

Darauf aufbauend folgt die dritte Ebene im Modell der strukturierten Dokumente (Formatebene), die sich auf die Formatierung der bereits strukturierten Inhalte bezieht. Das zu den Textelementen passende Format kann aus einer separaten Formatvorlage, auch Stylesheet genannt, erzeugt werden.

Selbsttestaufgabe 1.6

Was versteht man unter dem Begriff „Glatter Text"? Wozu benötigt man in diesem Zusammenhang eine Zeichencodierung?

Sogenannter „Glatter Text" ist eine Folge von elementaren Texteinheiten wie Buchstaben, Ziffern, Interpunktionszeichen und anderen Zeichen. Er enthält keinerlei Formatierungsangaben oder sonstige spezielle Steuerzeichen der Textcodierung oder Markierungen bezüglich der Strukturauszeichnung.

Eine Zeichencodierung gibt die Vorschrift dazu an, wie diese Zeichen in Bits und Bytes umgesetzt werden.

Selbsttestaufgabe 1.7

Welches sind die Bestandteile des im Lehrveranstaltungstext beschriebenen generischen Zeichensatzmodells? Erläutern Sie diese kurz in eigenen Worten.

Die einzelnen Bestandteile des Codierungsmodells sind:

- ein abstrakter Zeichensatz,
- eine Codetabelle,
- ein Codierungsformat,
- ein Codierungsschema und
- eine Übertragungssyntax.

Die obere Ebene des Codierungsmodells ist der abstrakte Zeichensatz. Dieser besteht aus einer Menge sogenannter abstrakter Zeichen. Es handelt sich hierbei um Buchstaben, Ziffern, Interpunktionszeichen, Akzente, grafische Symbole etc.

An nächster Stelle des Codierungsmodells steht die Codetabelle, die den abstrakten Zeichen im Zeichensatz eine Codeposition zuweist. Eine Codeposition ist immer eine natürliche Zahl, also größer oder gleich o. Der Code-Raum kann jedoch auch Zahlen enthalten, die keine zulässigen Codepositionen sind. Codepositionen werden üblicherweise in Hexadezimalnotation angegeben.

Mithilfe des Codierungsformats für eine Codetabelle werden die Bitrepräsentationen für die Codeposition festgelegt. Dazu gibt es eine Code-Einheit, die in der Regel aus acht oder sechzehn Bits besteht. Durch das Codierungsformat werden dann die Positionen eines Code-Raums in Sequenzen von Code-Einheiten und somit in Bitmuster abgebildet.

Ein Codierungsschema wird nun dazu genutzt zu einem Codierungsformat zusätzlich festzulegen, wie die Codierungseinheiten in Bytefolgen zu serialisieren sind.

Ist die Codierungseinheit eines Codierungsformats selbst acht Bits lang, so ist nichts mehr festzulegen und das Codierungsschema ist mit dem Codierungsformat identisch. So kann also UTF8 sowohl als Codierungsformat als auch als Codierungsschema bezeichnet werden.

Zur Abgrenzung von Markup und Inhalt werden bestimmte Funktionszeichen eingesetzt, die dann außerhalb ihrer syntaktischen Rolle weder im Markup selbst noch im Inhalt des Klartextes vorkommen dürfen. In XML ist < ein solches Funktionszeichen, das den Anfang eines Tags charakterisiert. Das Zeichen < darf deswegen im Inhaltstext eines XML- Dokuments nicht vorkommen.

Alle Rechte vorbehalten
© 2026 FernUniversität in Hagen
Fakultät für Mathematik und Informatik