

SGML

Grundkonzeption und Zusammenhang mit XML

Fallstudie im Fach **Betriebsinformatik**

Gutachter:

Prof. Dr. Uwe Kern

vorgelegt der



von:

Christian Kleyer
Tonstr. 13
45359 Essen

Matrikelnr. 14087

Daniel Oyen
Töpferweg 3
47665 Sonsbeck

Matrikelnr. 16025

Svend Reuse
Schinkelstr. 34
44801 Bochum

Matrikelnr. 16595

Essen, den 25.05.2001

Inhaltsverzeichnis

Inhaltsverzeichnis	I
Abbildungsverzeichnis	III
Abkürzungsverzeichnis.....	IV
1. Darstellung des Themas und Gang der Untersuchung	1
2. Grundkonzeption SGML	3
2.1. Definition des Markups	3
2.2. Definition des SGML-Standards	5
2.3. Historie von SGML	7
2.4. Aufbau von SGML Dokumenten	11
2.4.1. SGML Declaration	11
2.4.2. Document Type Definition.....	12
2.4.3. Dokumenteninstanz	14
2.5. Grundbestandteile von SGML.....	14
2.5.1. Elemente	14
2.5.2. Attribute.....	15
2.5.3. Entities.....	16
2.5.4. Document Style Semantics and Specification Language (DSSSL) ...	17
2.6. SGML-Werkzeuge	18
3. Wirtschaftliche Bedeutung von SGML	20
3.1. Wirtschaftliche Vorteile	20
3.2. Anwendung im Verlagswesen.....	23
3.3. Anwendung in der Industrie	24
3.4. Weltweiter Datenaustausch	25

4. SGML – die Mutter aller Sprachen	26
4.1. HTML.....	27
4.2. Extensible Markup Language – XML.....	28
5. Fazit und Ausblick für die Zukunft	32
<i>Literatur</i>	34
<i>Literaturquellen</i>	34
<i>Internetquellen</i>	35

Abbildungsverzeichnis

Abbildung 1: Entwicklungshistorie von SGML-basierten Sprachen26

Abbildung 2: Gegenüberstellung von HTML und XML.....29

Abbildung 3: XML-basierte Markup-Sprachen31

Abbildung 4: Zusammenhang zwischen SGML und XML32

Abkürzungsverzeichnis

AAP	=	Association of American Publishers
AG	=	Aktiengesellschaft
ANSI	=	American National Standards Institute
BMW	=	Bayerische Motorenwerke
bzw.	=	beziehungsweise
CA	=	California
CALS	=	Continuous Acquisition and Life-Cycle Support Systems
CDF	=	Channel Definition Format
CML	=	Chemical Markup Language
CSS	=	Cascading Style Sheets
d.h.	=	das heißt
DoD	=	US Department of Defense
DSSSL	=	Document Style Semantics and Specification Language
DTD	=	Document Type Definition
EADS	=	European Aeronautic Defense and Space Company
EDV	=	Elektronische Datenverarbeitung
ERP	=	Enterprise Ressource Planning
et al.	=	et altera
etc.	=	et cetera
f.	=	folgende
ff.	=	fortfolgende
GCA	=	Graphic Communications Association
GmbH	=	Gesellschaft mit beschränkter Haftung
GML	=	Generalized Markup Language

HTML	=	Hypertext Markup Language
HTTP	=	Hyper Text Transfer Protocol
IBM	=	Industrial Business Machines
IDC	=	International Data Corporation
ISO	=	International Standardization Organization
IT	=	Informationstechnologie
MP3	=	MPEG 2.5 Audio Layer III
MPEG	=	Motion Pictures Expert Group
o. J.	=	ohne Jahrgang
OASIS	=	Open Access Same-time Information System
OFX	=	Open Financial Exchange
PCDATA	=	Parsed Character Data
PDA	=	Personal Digital Assistants
RTF	=	Rich Text Format
S.	=	Seite
SGML	=	Standard Generalized Markup Language
SQL	=	Structured Query Language
STAGO	=	start-tag open
SVG	=	Scalable Vector Graphics
SyncML	=	Synchronisation Markup Language
TAGC	=	tag close
TCP/IP	=	Transmission Control Protocol / Internet Protocol
UMTS	=	Universal Mobile Telecommunications System
US	=	United States
vgl.	=	vergleiche

WML	=	Wireless Markup Language
WWW	=	World Wide Web
WYSIWYG	=	What you see is what you get
XHTML	=	Extended Hypertext Markup Language
XLS	=	Extensible Stylesheet Language
XML	=	Extensible Markup Language
z.B.	=	zum Beispiel

1. Darstellung des Themas und Gang der Untersuchung

Fortschreitende Vernetzung und Technisierung der Welt führen zu einem rasanten Wachstum des menschlichen Wissens, des weltweiten Informationsaustausches und letztlich auch des Datenaufkommens. Durch die Implementierung neuer Techniken wie UMTS¹ oder Anwendungen wie E-Commerce- und ERP²-Systeme wird das Datenwachstum noch weiter angeheizt werden. McKinsey erwartet, dass sich das Datenvolumen der Fortune 2000 Unternehmen jährlich verdoppelt und IDC³ prognostiziert ein jährliches Datenwachstum in europäischen Unternehmen von über 70% bis ins Jahr 2003. Dies wäre dann schon ein Datenvolumen von knapp 600.000 TeraBytes bzw. knapp das Vierfache des heutigen Datenvolumens.⁴ Weitere Studien schätzen, dass die entwickelten Länder etwa 20% ihres Bruttoinlandsproduktes für die Generierung neuer Informationen ausgeben, dass sich das weltweit verfügbare Wissen alle fünf bis sieben Jahre verdoppelt und sich etwa 90% aller Informationen in Dokumenten und nicht in Datenbanken befinden.⁵

Informationshaltung in Form traditioneller Dokumente bringt viele Nachteile mit sich. Wesentlich ist die Unterbrechung des Informationsflusses, sobald der Inhalt erst einmal in gedruckter Form vorliegt. Eine weitere Bearbeitung ist ausgeschlossen, da eine Verknüpfung zu einer elektronischen Datenbank fehlt. Zudem werden Informationen zunächst gesammelt, sortiert und organisiert, bevor diese letztlich als Papierdokument präsentiert werden. Jeder Schritt in diesem Prozess kostet Zeit und birgt die Gefahr einer fehlerhaften Informationsweitergabe, so dass das Risiko fehlerhafter Daten stetig größer wird, je weiter das Ergebnis von der ursprünglichen Informationsquelle entfernt ist. Dies kann soweit gehen, dass der Inhalt eines Dokuments bereits als veraltet gilt, wenn es endlich in gedruckter Form vorliegt. In konventionellen Textverarbeitungs- und Redaktionssystemen verbringen die Autoren solcher Dokumente etwa 30% ihrer Zeit mit der Suche nach den richtigen Informationen und wei-

¹ Universal Mobile Telecommunications System.

² Enterprise Resource Planning.

³ International Data Corporation.

⁴ Vgl. Lobster Network Storage AG (Data Storage, 2000), S. 3.

⁵ Vgl. Arbortext (Getting started with SGML, 1992/1995).

tere 30% für die entsprechende graphische Aufbereitung dieser Daten. Zudem verändert sich die technologische Basis innerhalb weniger Jahre so gravierend, dass Unternehmen gezwungen sind, ihre Datenbestände sowohl für die interne Verarbeitung und Bereitstellung als auch für die externe Publikation in neuen Ausgabemedien ständig an die neuen Soft- und Hardwareerfordernisse anzupassen und aufzubereiten.⁶

„Der schnelle Zugriff auf Informationen – von jedem Ort, zu jeder Zeit und durch jede beliebige Person – ist seit langem eine Vision der IT-Welt.“⁷ Untrennbar verbunden mit dieser Vision ist die Forderung der Adressaten der Informationen, diese auf ihre eigenen persönlichen Anforderungen und Wünsche hin maßgeschneidert zur Verfügung gestellt zu bekommen. Die Informationsempfänger möchten sie unverzüglich ohne weitere Aufbereitung für eigene Zwecke nutzen können – unabhängig vom Ausgabemedium oder bestimmten Systemstandards.⁸

Die Zukunft eines Unternehmens wird also wesentlich davon abhängen, wie effizient es seine Informationen und Daten sammelt, verwaltet und auch nutzt. Als Konsequenz daraus haben sich viele Branchen zusammengeschlossen, um gemeinsam einheitliche Standards zur Datenhaltung und zum Datenaustausch festzulegen und zu entwickeln. Geschäftsprozesse sollen dadurch effizienter gestaltet und die globale Wettbewerbsfähigkeit der beteiligten Unternehmen gesteigert werden.⁹

Die Standard Generalized Markup Language (SGML) und die Extensible Markup Language (XML) sind Standards, die einen wesentlichen Teil einer ganzheitlichen Informationsmanagementstrategie zur Realisierung dieser Vision darstellen. Beide ermöglichen es, Daten nicht nur als Aneinanderreihung von Zeichen, sondern als Datenobjekte zu betrachten. So können die Informationen gespeichert, wiederverwendet, mit vielen Benutzern geteilt und in einer Datenbank gehalten werden.¹⁰ Eine

⁶ Vgl. Arbortext (Getting started with SGML, 1992/1995).

⁷ Ruettgers, M. C. (E-Infostructure, o.J.).

⁸ Vgl. Holloway, H. L. (Generic Coding and SGML, 1987), S. 1.

⁹ Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹⁰ Vgl. Arbortext (Getting started with SGML, 1992/1995);
vgl. Sperberg-McQueen, C.M./Burnard, L. (A Gentle Introduction to SGML, o.J.).

2.1. Definition des Markups

vollständige Hard- und Softwareunabhängigkeit sowie die Publikation der Daten auf den unterschiedlichsten Medien wird gewährleistet.¹¹

In den folgenden Kapiteln dieser Fallstudie wird zunächst die Idee und Entwicklung von SGML dargestellt. Ein Schwerpunkt wird dabei auf der konzeptionellen Gestaltung dieses Standards liegen. Im weiteren wird eine Beurteilung der wirtschaftlichen Bedeutung von SGML erfolgen, bevor abschließend Zusammenhänge zwischen SGML und seinem Quasi-Nachfolgestandard XML aufgezeigt werden.

2. Grundkonzeption SGML

2.1. Definition des Markups

Der Begriff „Markup“ lässt sich mit Auszeichnung übersetzen. Darunter versteht man – außer dem Inhalt – alle Mittel, die eine Interpretation eines Dokuments ermöglichen.¹² Markup-Languages bzw. Auszeichnungssprachen sind textbasierte Daten-Beschreibungs-Sprachen (Meta-Sprachen).¹³

Historisch bezeichnete Markup die handgeschriebenen Anmerkungen eines Autors oder Textdesigners an einer Textvorlage; man bezeichnet dies auch als „grafisches Markup“. Diese Notizen dienten als Anweisungen für den Setzer, wie bestimmte Passagen des Textes oder einzelne Wörter gedruckt werden sollten. Als nun durch die technologische Entwicklung die Formatierung und Drucklegung von Texten automatisiert wurde, bekam auch der Begriff Markup eine neue Bedeutung. Heutzutage spricht man von einem „generischen Markup“ und versteht darunter alle Arten von (Auszeichnungs-)Befehlen, die zur Übermittlung von Angaben zur Drucklegung, Formatierung und sonstigen Auszeichnung der Daten in ein elektronisches Doku-

¹¹ Hofmann, T./Raithelhuber, U. (SGML/XML, 1998).

¹² Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹³ Meyer, M. (XML – was ist es?, o.J.).

2.1. Definition des Markups

ment integriert werden.¹⁴ Diese Auszeichnungsbefehle werden „tags“ genannt und durch bestimmte Trennzeichen vom übrigen Text separiert, damit der Computer diese als solche erkennen und gemäß seiner Programmierung interpretieren kann.¹⁵ Generisches Markup meint, dass mit Hilfe einer sogenannten semantischen Auszeichnung einzelnen Textpassagen, Dateien, etc. bestimmte Bedeutungen und Relationen untereinander zugewiesen werden. Dies geht weit über die Vorgehensweise von üblichen WYSIWYG¹⁶-Textverarbeitungsprogrammen wie z.B. Word hinaus. Diese arbeiten intern lediglich mit grafischen Markups und verbergen diese vor dem Benutzer. Im Gegensatz dazu stehen von SGML abgeleitete Textsysteme wie Latex.¹⁷

Jedes WYSIWYG-Textverarbeitungsprogramm hat eine eigene nur für sich verständliche Auswahl an grafischen Markup-Befehlen zur Präsentation der Dokumente auf dem Bildschirm oder als Ausdruck. Diese Befehle verfügen jedoch nicht über die Fähigkeiten, um die Präsentation dieser Dokumente auf anderen Ausgabemedien wie z.B. CD-ROMs, Organizern oder Handys zu definieren. Dagegen beschreibt generisches Markup mittels semantischer „tags“ die Bedeutung einer Textpassage innerhalb eines Dokumentes, anstatt die physische Erscheinung eines Textes zu bestimmen. Kernidee ist hier die Trennung des Inhaltes eines Dokumentes und dessen Auszeichnung. Generisches Markup basiert auf der Struktur eines Dokumentes und benennt anhand von Auszeichnungen einzelne Elemente in diesem Dokument anstatt deren Erscheinung festzulegen. Diese benannten, mit einer interpretierbaren Bedeutung versehenen, Elemente, beispielsweise Kapitel oder Absätze, können später wieder identifiziert werden.

Wenn nun Präsentation und Struktur eines Dokuments voneinander getrennt werden, so ergeben sich bedeutende Vorteile.¹⁸

¹⁴ Vgl. Sperberg-McQueen, C.M./Burnard, L. (A Gentle Introduction to SGML, o.J.).

¹⁵ Vgl. Holloway, H. L. (Generic Coding and SGML, 1987), S. 3f.

¹⁶ What You See Is What You Get.

¹⁷ Meyer, M. (XML – was ist es?, o.J.).

¹⁸ Vgl. Arbortext (Getting started with SGML, 1992/1995).

2.2. Definition des SGML-Standards

- Generisches Markup ermöglicht eine vielfache Darstellung der selben Information auf verschiedenen Ausgabemedien bzw. in verschiedenen Formaten, ohne dass eine Neuformatierung der Daten notwendig wird.
- Software und Hardware können ausgetauscht werden, ohne dass eine (fehlerbehaftete) Datenübersetzung erforderlich wird.
- Dokumente können problemlos zwischen verschiedenen Systemen ausgetauscht werden.

Entsprechend dieser Vorteile bei der Bewältigung des Datenwachstums wächst die Bedeutung von Auszeichnungssprachen, die eine generische Auszeichnung ermöglichen, signifikant.

Eine Auszeichnungssprache besteht aus einer Sammlung von Auszeichnungsvereinbarungen, die für die Kodierung eines Textes benutzt werden. Sie muss definieren, welche Auszeichnungen erlaubt sind, welche Auszeichnung erforderlich ist, wie diese Auszeichnungen vom „Text“ zu unterscheiden sind und was diese Auszeichnungen bedeuten. SGML verfügt über die Möglichkeiten zur Erfüllung dieser Bedingungen.

2.2. Definition des SGML-Standards

SGML stellt einen internationalen Standard für die Definition einer Dokumentenauszeichnungssprache bzw. von Auszeichnungsbefehlen dar. Dabei ist SGML selbst keine Sprache zur Auszeichnung von Dokumenten, sondern stellt lediglich die Vorschriften bereit, um eine solche Markup- bzw. Auszeichnungssprache formal zu definieren. SGML zählt zu den Metasprachen.¹⁹ Sie ist 1986 von der International Standardization Organization (ISO) unter der Nummer ISO 8879-1986 zum Standard für die Beschreibung von Dokumenten festgelegt worden.²⁰ Die Beschreibung von

¹⁹ Vgl. Whatis?com (Stichwort SGML).

²⁰ Vgl. Hofmann, T./Raitelhuber, U. (SGML/XML, 1998).

2.2. Definition des SGML-Standards

Dokumenten umfasst die Definition, Identifikation und Benutzung der Struktur und des Inhalts von Dokumenten.²¹

Mittels SGML kann für jede erzeugte Dokumentenart eine hierarchische Struktur festgelegt werden, die anschließend jedem entsprechend generisch ausgezeichnetem Element (Titel, Absatz, Passage, etc.) seine vorherbestimmte Position im Dokument zuweist. Dabei unterstützt SGML unendlich viele Dokumentenarten, so dass für jede Art der Information und Verarbeitung, z.B. für einen Brief oder ein Memo, eine eigene Dokumentenart erzeugt werden kann.²²

Die Idee und das Ziel von SGML sind die oben beschriebene Trennung der Struktur des Inhalts eines Dokuments von seinem layoutorientierten Erscheinungsbild. Somit tritt bei der Formatierung von Dokumenten die logische und strukturelle Auszeichnung in den Vordergrund, während die Formatierung des typografischen Layouts an Bedeutung verliert. Der Nutzen aus dieser Vorgehensweise ist, dass diese Dokumente sowohl vollständig unabhängig von bestimmten Hard- und Softwaresystemen sind, als auch die Daten dieser Dokumente problemlos auf den unterschiedlichsten Ausgabemedien dargestellt werden können.²³

Arbortext vergleicht die Bedeutung des SGML Standards für Dokumente beispielsweise mit der Wichtigkeit der standardisierten Filmgeschwindigkeit für die Fotoindustrie. Dank diesem Standard können Verbraucher heutzutage weltweit jeden beliebigen Fotofilm kaufen, der mit der Norm „ISO 100“ gekennzeichnet ist. Es spielt dabei keine Rolle mehr, ob der Produzent der Fotokamera und der Produzent des Kamerafilms identisch sind. SGML soll ermöglichen, dass zukünftig Dokumente auf verschiedenen Computersystemen genauso problemlos bearbeitet werden können wie heute Fotofilme von verschiedenen Kameras akzeptiert werden.²⁴

²¹ Vgl. Meyer, M. (XML – was ist es?).

²² Vgl. Arbortext (Getting started with SGML, 1992/1995).

²³ Vgl. Whatis?com (Stichwort SGML) und Meyer, M. (XML – was ist es?).

²⁴ Vgl. Arbortext (Getting started with SGML, 1992/1995).

2.3. Historie von SGML

Die Anfänge bzw. Wurzeln von SGML reichen zurück bis in die 60er Jahre des vergangenen Jahrhunderts, als die ersten Ideen für ein generisches Befehlskonzept aufkamen und wenige Jahre später die Generalized Markup Language (GML) entwickelt wurde.²⁵ Diese gilt als Vorläufer von SGML und wurde federführend von Charles Goldfarb bei IBM entwickelt.²⁶

Ursprünglich beinhalteten elektronische Dokumente Steuerungsbefehle und Makros, die bewirkten, dass ein Dokument in einer bestimmten Art und Weise formatiert wurde. Mit dem Aufkommen des generischen Befehlskonzepts in den späten 1960ern begann der Gebrauch beschreibender „tags“, wie z.B. „heading“ anstatt „format-17“.²⁷ Vorherrschende Meinung ist heute, dass der Anstoß für die Entwicklung eines generischen Befehlskonzepts 1967 von William Tunnicliffe, dem damaligen Vorsitzenden der Graphic Communications Association (GCA) Composition Committee, ausgegangen ist.²⁸ Er hielt im September eben diesen Jahres bei einem Treffen des Canadian Government Printing Office einen Vortrag mit dem Thema „The Separation of the Information Content of Documents from their Format.“. Unabhängig von William Tunnicliffe publizierte zur gleichen Zeit der New Yorker Buchdesigner Stanley Rice Artikel zum Thema „Standardized Editorial Structures“. Seine Vision war, dass parametrisierte Layoutbefehle auf der inhaltlichen Struktur einer Publikation beruhen sollten.²⁹ Schließlich erkannte der Direktor der GCA, Norman Scharpf, die Bedeutung dieser Trends und gründete das „generic coding project“ im Composition Committee. Dieses Projekt entwickelte das „GenCode(R) Concept“. Eine wesentliche Erkenntnis hierbei war, dass verschiedene Dokumentenarten auch verschiedene generische Befehle benötigen würden und das kleinere Dokumenttypen in größere integriert werden könnten. Schließlich entstand aus diesem Projekt das

²⁵ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

²⁶ Vgl. Hofmann, T./Raitelhuber, U. (SGML/XML, 1998).

²⁷ SGML User's Group (A Brief History of the Development of SGML, 1990).

²⁸ Vgl. Goldfarb, C. F. (The Roots of SGML – A Personal Recollection, 1996).

²⁹ Vgl. Goldfarb, C. F. (The Roots of SGML – A Personal Recollection, 1996).

2.3. Historie von SGML

GenCode Committee, welches später eine bedeutende Rolle in der Entwicklung des SGML-Standards spielen sollte.³⁰

Charles F. Goldfarb, der auch als Vater von SGML und als Großvater dessen Abkömmlings HTML³¹ gilt³², hatte ursprünglich nichts mit Computern zu tun. Er hatte in Harvard Jura studiert und arbeitete danach als Rechtsanwalt in Boston. Eines seiner Hobbies war der Entwurf von Streckenbeschreibungen für Rallye Sportwagen. Ein Freund von ihm fand, dass diese Beschreibungen Computerprogrammen gleichen würden. Daher wagte Goldfarb 1967, eine Anstellung bei IBM anzunehmen, obwohl er keine Vorstellung von Computerprogrammen hatte. Angeblich reizte es ihn, zu erfahren, warum ihm IBM Geld für den Entwurf von Rennanweisungen zahlen würde.³³ Anfang 1969 überzeugte ihn IBM, dem eigenen Cambridge Scientific Center beizutreten und dort das Forschungsprojekt „Integrated Law Office Information Systems“ zu leiten. Zusammen mit Ed Mosher und Raymond Lorie entwickelte er dort die Generalized Markup Language (GML) als ein Werkzeug, dass verschiedenen Systemen zur Textbearbeitung, -formatierung und Informationsbereitstellung den Zugriff auf die gleichen Daten bzw. Dokumente ermöglichen sollte.³⁴ GML basierte dabei auf den wegweisenden Ideen von Rice und Tunnicliffe und führte das Konzept eines formal definierten Dokumententyps mit einer ausdrücklich dazugehörigen Elementstruktur ein. Als IBM die Bedeutung des Projekts erkannte, wurde der Projektname in „Integrated Text Processing“ geändert und der Stellenwert des Projekts deutlich angehoben.³⁵ GML-Auszeichnungen konnten eindeutig Dokumentenbereiche wie Kapitel, wichtige und unwichtige Abschnitte beschreiben. Dies brachte für die Erzeuger elektronischer Dokumente einen großen Vorteil, da sie sich keine Gedanken mehr um das Seitenlayout, den Zeilenabstand oder weitere Spezifikationen in Zusammenhang mit der Erscheinung bzw. Ausgabe der Dokumente machen brauchten. Solche Angaben hatte bis dahin IBMs Druckformatsprache SCRIPT von

³⁰ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

³¹ Hypertext Markup Language.

³² Vgl. Goldfarb, C. F. (The Roots of SGML – A Personal Recollection, 1996).

³³ Vgl. Goldfarb, C. F. (The Roots of SGML – A Personal Recollection, 1996).

³⁴ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

³⁵ Vgl. Goldfarb, C. F. (The Roots of SGML – A Personal Recollection, 1996).

2.3. Historie von SGML

Ihnen gefordert.³⁶ In den folgenden Jahren wurden wesentliche Teile von GML in die Publikationssysteme von IBM und weiteren Herstellern integriert, so dass GML eine hohe Akzeptanz erreichte. Goldfarb setzte seine Forschungen hinsichtlich Dokumentenstrukturen und die Entwicklung weiterer Konzepte fort. Diese waren zwar nicht Teil von GML sollten aber später Bestandteil von SGML werden.³⁷ Schließlich verständigten sich die GCA und Goldfarb 1978, die bisher unabhängigen Forschungen und Entwicklungen zukünftig gemeinsam fortzusetzen. Damit wurde der Grundstein für die Entwicklung des SGML-Standards gelegt.³⁸

Der Ausschuss für Informationsverarbeitung des American National Standards Institute (ANSI) etablierte in 1978 einen entsprechenden Ausschuss für diesen Zweck. Neben Mitgliedern wie Norman Scharpf wurde Charles F. Goldfarb eingeladen, diesem Ausschuss beizutreten und schließlich auch das Projekt zur Entwicklung eines Standards für Auszeichnungssprachen basierend auf GML zu leiten. Der GCA-GenCode-Ausschuss unterstützte diese Bemühungen und stellte eine Gruppe hochkarätiger Mitarbeiter für die Verwirklichung dieses Projektes zur Verfügung.³⁹

Der erste funktionierende Entwurf des SGML Standards wurde 1980 publiziert. Drei Jahre später gab es bereits sechs funktionierende Entwürfe und die GCA empfahl diese als Industriestandard (GCA 101-1983). Zu den ersten bedeutenden Anwendern dieses Standards zählten der US Internal Revenue Service (IRS) und das US Department of Defense (DoD). In 1984 wurden aufgrund des positiven Feedbacks auf den CGA-Standard drei weitere funktionierende Entwürfe vorgestellt und das Projekt, mittlerweile von der ISO und ANSI autorisiert, reorganisiert. Es begannen nun regelmäßige internationale Treffen unter der Bezeichnung „ISO/IEC JTC1/SC18/WG8“ unter der Leitung von James Mason vom US Oak Ridge National Laboratory. Parallel dazu wurde die Arbeit im ANSI Ausschuss unter der Bezeichnung „X3V1.8“ unter Leitung von William Davis von SGML Associates und mit Unterstützung vom GCA GenCode Ausschuss unter Sharon Adler von IBM fortge-

³⁶ Vgl. Whatis?com (Stichwort GML).

³⁷ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

³⁸ Vgl. Goldfarb, C. F. (The Roots of SGML – A Personal Recollection, 1996).

³⁹ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

2.3. Historie von SGML

setzt. Koordiniert wurden diese verschiedenen Aktivitäten zwischen ISO und ANSI durch Charles F. Goldfarb, der weiterhin die technische Leitung inne hatte und beiden Ausschüssen als Projektleiter diente.⁴⁰

Im Jahr 1985 wurde ein erster Entwurfsvorschlag für den internationalen Standard publiziert. Kurz darauf wurde die internationale SGML Users' Group in Großbritannien durch Joan Smith gegründet. Im Oktober 1985 wurde dann der Entwurf für den internationalen Standard präsentiert, der auch vom Büro für öffentliche Publikationen der Europäischen Union akzeptiert wurde. Ein Jahr später erfolgte die Publikation der abschließend überarbeiteten Version des neuen Standards unter der bekannten Kennzeichnung ISO 8879-1986. Benutzt wurde dazu ein von Anders Berglund vom European Particle Physics Laboratory entwickeltes SGML-System.⁴¹

Eine der ersten großen SGML-Anwendungen war die Dokumentationskomponente des Continuous Acquisition and Life-Cycle Support Systems (CALS), der ursprünglichen Computer-aided Acquisition and Logistic Support Initiative, des US DoD. Die Entwicklung dieser Komponente begann im Februar 1987 und führte zu der Publikation des Militärstandards MIL-M-28001 für den Textaustausch im Februar des darauffolgenden Jahres.⁴² CALS ist als großes Langzeitinformationssystem initiiert worden. Da das US DoD eine Vielzahl an Gütern und Diensten von einer Vielzahl an Zulieferern und Dienstleistern empfängt, muss ständig eine Unmenge technischer Informationen verwaltet werden. Die heutigen Waffensysteme sind technologisch äußerst komplex und haben meistens einen Lebenszyklus von mehr als 20 Jahren, so dass eine gewaltige Datenmenge zur Verwaltung und Instandhaltung dieser Systeme benötigt wird. Zudem müssen die vorhandenen Daten auch auf zukünftigen Systemplattformen noch verfügbar und nutzbar sein. CALS soll helfen, die Unterhalts- und Instandhaltungskosten für militärisches Gerät zu senken.⁴³ SGML hilft, die Zukunftssicherheit der Daten zu gewährleisten. Weitere bekannte und große Anwender von

⁴⁰ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

⁴¹ Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

⁴² Vgl. SGML User's Group (A Brief History of the Development of SGML, 1990).

⁴³ Vgl. Arbortext (Getting started with SGML, 1992/1995).

SGML sind beispielsweise die European Aeronautic Defence and Space Company (EADS) oder die Association of American Publishers (AAP).

2.4. Aufbau von SGML Dokumenten

Ein SGML-Dokument besteht in der Regel aus einem Prolog und einer Dokumenteninstanz. Der Prolog ist der formale Teil eines Dokuments und enthält die sogenannte „SGML Declaration“ und eine Document Type Definition, kurz DTD genannt. Bei der Dokumenteninstanz handelt es sich um den eigentlichen Inhalt des Dokuments.⁴⁴

2.4.1. SGML Declaration

Jede Programmiersprache basiert auf einer Syntax, d.h. auf Regeln und Definitionen, welche unter anderem die Bedeutung einzelner Zeichen und Funktionen bestimmen.⁴⁵ Die Syntax von SGML wird in der „SGML Declaration“ festgelegt und besteht aus einem abstrakten und einem konkreten Teil. Die abstrakte Syntax legt die Definition und Konstruktion der Auszeichnungsbefehle fest. Darüber hinaus existiert eine anschauliche, konkrete Syntax, die vorschreibt, wie ein Dokument kodiert wird.⁴⁶ Sie ordnet einen definierten Befehl einer bestimmten Zeichenfolge zu, d.h. sie bestimmt, welches Zeichen einen bestimmten Befehl repräsentiert. Beispielsweise wird ein eröffnender Auszeichnungsbefehl, also ein „start-tag“, von einem „start-tag open“ (STAGO) und einem „tag close“ (TAGC) umschlossen. Diese Namen sowie die Bedeutung des Befehls werden in der abstrakten Syntax definiert. Die konkrete Syntax legt dann fest, dass die Zeichen „<“ als STAGO und das Zeichen „>“ als TAGC interpretiert werden.⁴⁷ Darüber hinaus kann mit der konkreten Syntax auch

⁴⁴ Vgl. Sperberg-McQueen, C.M./Burnard, L. (A Gentle Introduction to SGML, o.J.).

⁴⁵ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 121.

⁴⁶ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1988), S. 59.

⁴⁷ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 132; Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1988), S. 68.

2.4. Aufbau von SGML Dokumenten

bestimmt werden, welcher Zeichensatz benutzt werden soll, welche Zeichen nicht im Dokument vorkommen dürfen, welche Zeichen in den Befehlsnamen benutzt werden und wie lang diese Namen sein dürfen.⁴⁸

Die konkrete Syntax kann im Grunde frei definiert bzw. angepasst werden. Die ISO-Norm 8897 enthält jedoch eine konkrete Syntax als allgemeingültige Referenz. Jedes SGML-fähige System muss diese Referenzsyntax unterstützen und in der Lage sein, Dokumente mit einer selbstdefinierten Syntax in die Standardsyntax zu konvertieren.⁴⁹ In der Praxis soll diese Konvertierung jedoch nicht automatisch erfolgen. Vielmehr soll der Nutzer für die Umsetzung zuständig sein und die entsprechenden Vorgaben dafür machen. Da die Syntax sowohl für Computer als auch für den Menschen lesbar ist, können die notwendigen Konvertierungstabellen schnell angelegt werden. Aufgrund der damit verbundenen Fehleranfälligkeit wird allerdings empfohlen, auf die sorgfältig geprüften, öffentlich verfügbaren Syntaxen zurückzugreifen.⁵⁰

2.4.2. Document Type Definition

In einer DTD werden die Regeln für den Aufbau von SGML-Dokumenten geliefert. Dementsprechend beginnt jedes SGML-Dokument mit einer DTD.⁵¹ Dort wird definiert, welche „tags“ es in einem bestimmten Dokumententyp geben kann, welche Attribute ein „tag“ enthalten kann, von welcher Art die Werte dieser Attribute sein müssen, welche Elemente verwendet werden dürfen und wie diese miteinander verschachtelt werden müssen.⁵² Ferner legt eine DTD fest, ob bestimmte Auszeichnungen weggelassen werden können und welche Entities⁵³ benutzt werden dürfen.⁵⁴

⁴⁸ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 121; Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1988), S. 62ff.

⁴⁹ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1988), S. 59.

⁵⁰ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1988), S. 72f.

⁵¹ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1988), S. 20.

⁵² Vgl. Dünhölter, K. (Das Web automatisieren mit XML, 1998).

⁵³ Siehe Abschnitt 2.5.3.

⁵⁴ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 33.

2.4. Aufbau von SGML Dokumenten

Dementsprechend unterscheiden sich die DTDs von verschiedenen Dokumententypen wie z.B. einem Memo, einem Brief oder einem Telefonbuch.⁵⁵ So wie man ein Buch mit Telefonnummern und Namen sofort als Telefonbuch erkennt, kann ein SGML-Dokument über seine „tags“ als solches identifiziert werden. Eine DTD kann daher auch als Grammatik einer Auszeichnungssprache bezeichnet werden. DTDs stellen in diesem Zusammenhang mächtige Werkzeuge zur Standardisierung dar, denn sie entscheiden über die Weiterverarbeitung der Dokumente durch Programme.⁵⁶ Für Dokumente, die den gleichen Regeln unterliegen, können Anwendungen geschaffen werden, die die Informationen aus diesen auslesen und automatisch weiterverarbeiten.⁵⁷ Deswegen ist es sehr wichtig, genau zu spezifizieren, welche DTD man nutzt, wenn man über SGML spricht.⁵⁸ Eine DTD wird auch als das Herz einer SGML-Anwendung bezeichnet und ist grundsätzlich in SGML geschrieben.⁵⁹

Werden Dokumente mit einer DTD zusammen versendet, so kann jeder Empfänger, der über einen „DTD reader“ bzw. SGML-Parser⁶⁰ verfügt, diese Dokumente verarbeiten und so auf einem Ausgabemedium darstellen, wie es vom Absender der Dokumente gewünscht wird. Der Parser liest die DTD und stellt dann in der beschriebenen Art und Weise das Dokument dar. Ein einziger SGML-Parser kann eine Vielzahl unterschiedlicher Dokumententypen mit einer Unmenge verschiedener Auszeichnungsbefehle handhaben, sofern er die „tags“ kennt.⁶¹

Document Type Definitions haben gegenüber bekannten Datenformaten wie etwa einer Word- oder Excel-Datei zwei entscheidende Vorteile: Sie sind unabhängig von Systemplattformen und befinden sich nicht im Besitz einer einzelnen oder mehrerer Firmen.⁶² Durch die Entwicklung der Document Type Definitions konnten die Probleme, die sich zuvor aufgrund weniger Standard-„tags“ und Dokumentenstrukturen ergeben hatten, gelöst werden. Jede Firma, Benutzergruppe, etc. kann nun ihre eige-

⁵⁵ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 25.

⁵⁶ Vgl. Martiné, A. (Die Grammatik des Webs, 2001).

⁵⁷ Vgl. Dünhölter, K. (Das Web automatisieren mit XML, 1998).

⁵⁸ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 33.

⁵⁹ Vgl. Arbortext (Getting started with SGML, 1992/1995); vgl. van Herwijnen, E. (Practical SGML, 1994), S. 25.

⁶⁰ Siehe Abschnitt 2.6.

⁶¹ Vgl. Whatis?com (Stichwort DTD).

⁶² Vgl. Martiné, A. (Die Grammatik des Webs, 2001).

2.5. Grundbestandteile von SGML

ne Auszeichnungssprache entwickeln und weiterhin die vorhandenen SGML-Werkzeuge und -Produkte nutzen.⁶³

Die Markup-Deklarationen einer DTD für Briefe könnte beispielsweise wie folgt aussehen:⁶⁴

```
<!ELEMENT brief (anrede, adresse)>
<!ELEMENT anrede (#PCDATA)>
<!ELEMENT adresse (#PCDATA)>
```

Durch die Deklaration wird festgelegt, dass in dem benannten Dokumententyp „Brief“ die Elemente Anrede und Adresse vorkommen müssen. Die Abkürzung PCDATA steht für „Parsed Character Data“ und bedeutet, dass in den Elementen lediglich reiner Text vorkommen darf.

2.4.3. Dokumenteninstanz

Die Dokumenteninstanz stellt das eigentliche Dokument dar. Sie enthält die Inhalte im Textformat sowie die entsprechenden Auszeichnungen, welche im folgenden Abschnitt vorgestellt werden. Darüber hinaus enthält sie einen Verweis auf die DTD, falls letztere nicht in vollem Umfang innerhalb der Instanz enthalten ist.⁶⁵

2.5. Grundbestandteile von SGML

2.5.1. Elemente

Ein SGML-Dokument besteht aus einer Vielzahl von eindeutig identifizierbaren Elementen. Diese enthalten entweder einen Text oder weitere, hierarchisch organisier-

⁶³ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 32.

⁶⁴ Martiné, A. (Die Grammatik des Webs, 2001).

⁶⁵ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 26.

2.5. Grundbestandteile von SGML

te Subelemente.⁶⁶ Jedem Elementtyp wird in der DTD ein Name – ein sogenannter „generic identifier“ – zugewiesen. Diese generic identifier bilden die „tags“, sobald sie mit Trennzeichen versehen werden.⁶⁷ Ein einzelnes Element besteht nun aus einem „start-tag“, dem eigentlichen Inhalt des Elements und einem „end-tag“.⁶⁸

Zur Verdeutlichung soll ein Elementtyp betrachtet werden, der den Titel eines Dokuments darstellt. Der Elementname bzw. der generic identifier lautet „title“. Das gesamte Element sieht dann wie folgt aus:

```
<TITLE> Titel des Dokuments </TITLE>
```

2.5.2. Attribute

Einem Element können mehrere Attribute zugewiesen werden. Ein Attribut ist ein Parameter und enthält eine beschreibende Information über das Element, ohne das diese zum Inhalt des Elements gehört.⁶⁹ Attribute und deren Ausprägungen werden in der DTD deklariert. Sie werden immer dem „start-tag“ beigefügt und bestehen aus dem Namen des Attributes, einem „=-“-Zeichen und der zugewiesenen Ausprägung.⁷⁰

Das Element „Body“, d.h. der Textkörper eines Dokuments, könnte z.B. durch die Attribute „Format“ und „Status“ näher bestimmt werden:

```
<BODY FORMAT="standard" STATUS="public">
```

Hier wird dem Textkörper ein Standardformat zugewiesen; gleichzeitig wird festgelegt, dass sein Inhalt öffentlich lesbar sein soll.

⁶⁶ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1998), S. 15.

⁶⁷ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1998), S. 20.

⁶⁸ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 35.

⁶⁹ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1998), S. 138; Sperberg-McQueen, C.M./Burnard, L. (A Gentle Introduction to SGML, o.J.).

⁷⁰ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 37.

2.5.3. Entities

Eine dritte Art von Auszeichnungsbefehlen sind die Entities. Eine Entity ist in der ISO 8879 definiert als „a collection of characters that can be referenced as a unit“.⁷¹ Durch Entities kann innerhalb eines SGML-Dokuments auf bestimmte Objekte verwiesen werden. Man unterscheidet „general entities“, die auf einen vorher definierten Text verweisen, „character entities“, die spezielle Symbole und Sonderzeichen darstellen sollen, und „external entities“, die auf externe Dokumente oder auch Grafiken verweisen. Darüber hinaus existieren „parameter entities“, die als Variablen in der DTD genutzt werden.⁷²

Eine Entity muss zunächst definiert werden. Diese „entity declaration“ erfolgt entweder in der DTD oder innerhalb der Document Type Declaration, welche sich am Anfang des SGML-Dokuments befindet. Mit ihr werden der Entity-Name sowie die Inhalte vorgegeben, die später ausgegeben werden sollen.⁷³ Die Definition einer general entity könnte z.B. wie folgt aussehen:

```
<!ENTITY SGML "Standard Generalized Markup Language">
```

Um diese Entity zu nutzen, ist eine sogenannte „entity reference“ notwendig.⁷⁴ Die Referenz besteht wieder aus dem Namen der Entity, der mit entsprechenden Trennzeichen – einem „&“ zu Anfang und einem Semikolon am Ende – umschlossen wird.⁷⁵ Bezogen auf das obige Beispiel sieht die „entity reference“ wie folgt aus:

```
&SGML;
```

Innerhalb eines Dokumentes kann diese Referenz einfach in den normalen Text integriert werden:

⁷¹ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1998), S. 74.

⁷² Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 39 f.

⁷³ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1998), S. 75.

⁷⁴ Vgl. Bryan, M. (SGML - An Author's Guide to the Standard Generalized Markup Language, 1998), S. 75.

⁷⁵ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 41.

2.5. Grundbestandteile von SGML

<p>Im zweiten Kapitel wird die &SGML; erklärt.</p>

Der SGML-Parser⁷⁶ sorgt dafür, dass die entity reference durch den Inhalt der Entity ersetzt wird, so dass bei der Ausgabe des Textes der vollständige Satz zu lesen ist:

Im zweiten Kapitel wird die Standard Generalized Markup Language erklärt.

Das gleiche Prinzip lässt sich bei der Einbindung von Sonderzeichen, Grafiken oder Texten aus externen Dateien anwenden.

2.5.4. Document Style Semantics and Specification Language

(DSSSL)

Die Abkürzung DSSSL – gesprochen „Dissel“⁷⁷ – steht für Document Style Semantics and Specification Language und ist ein internationaler Standard für die Bearbeitung von SGML-Dokumenten.⁷⁸ Da SGML selbst mittels der konsequenten Trennung von Inhalt und Layout eines Dokuments lediglich dessen logische Struktur festlegt, war die Entwicklung einer entsprechenden Stilsprache notwendig.⁷⁹ Eine Stilsprache entscheidet, wie ein derart strukturiertes Dokument grafisch auf einem Ausgabemedium dargestellt, eventuell in einen anderen Standard wie z.B. HTML umgewandelt oder in einer anderen Art und Weise bearbeitet wird.⁸⁰ DSSSL besteht dazu aus mehreren voneinander unabhängigen Komponenten, die je nach Bedarf für die Festlegung einer bestimmten DSSSL-Definition angesprochen werden können. Im einzelnen verfügt DSSSL über Standards für die Stilsprache, die Flow-Objekte, die Transformationssprache, das Dokumentenmodell und den Abfragemechanismus.⁸¹

⁷⁶ Vgl. Kapitel 2.6.

⁷⁷ Reibold, H. (Stilsprache für SGML, 2001).

⁷⁸ Vgl. Whatis?com (Stichwort DSSSL).

⁷⁹ Vgl. Prescod, P. (Introduction to DSSSL, 1997).

⁸⁰ Vgl. Whatis?com (Stichwort DSSSL).

⁸¹ Vgl. Whatis?com (Stichwort DSSSL).

2.6. SGML-Werkzeuge

Die Funktionen von DSSSL führt ein sogenannter DSSSL-Prozessor wie beispielsweise Jade (für Formate wie HTML, RTF⁸²) oder Seng (für Java) aus.⁸³ Dieser erzeugt dazu zunächst ein SGML-Dokument samt zugehöriger DTD und dem entsprechenden Style Sheet. Der Standard für eine Stilsprache stellt eine leistungsfähige Sprache für die Beschreibung und Formatierung von SGML-Dokumenten zur Verfügung. Das Layout eines Dokuments bestimmt DSSSL über die sogenannten Flow-Objekte.⁸⁴ Dabei handelt es sich um ausgezeichnete Elemente wie Kapitel, Absätze, Hyperlinks oder Bilder. Diesen kann der Entwickler – wie in „normalen“ Textverarbeitungsprogrammen – bestimmte Charakteristika zuweisen, wie z.B. Farbe, Schriftart oder Ziel eines Hyperlinks. Die Transformationssprache ist eine Standardsprache für die Umwandlung eines SGML-Dokuments in einen anderen Dokumententyp wie XML oder HTML. Dies geschieht durch die Transformation der zugehörigen DTD in die entsprechende des anderen Dokumententyps.⁸⁵ Mit Hilfe des Dokumentenmodells kann betrachtet werden, wie ein Dokument im einzelnen organisiert ist. Der von DSSSL zur Verfügung gestellte Abfragemechanismus (Query Language) ermöglicht das Durchsuchen von Dokumenten nach Inhalten, ähnlich wie Datenbankabfragen (SQL⁸⁶).⁸⁷ Voraussetzung hierfür ist aber, dass der DSSSL Prozessor diese Abfragefunktion auch unterstützt.⁸⁸

2.6. SGML-Werkzeuge

Für den Umgang mit SGML wurden verschiedene Werkzeuge entwickelt. Zunächst existieren die sogenannten Editoren. Dabei handelt es sich um Programme, die dem Benutzer helfen, Dokumente SGML-konform zu erstellen.⁸⁹ Sie gewährleisten unter anderem, dass Elemente nur an den von der DTD vorgegebenen Stellen im Doku-

⁸² Rich Text Format.

⁸³ Vgl. Reibold, H. (Stilsprache für SGML, 2001).

⁸⁴ Vgl. Prescod, P. (Introduction to DSSSL, 1997).

⁸⁵ Vgl. Martiné, A. (Die Grammatik des Webs, 2001) und Prescod, P. (Introduction to DSSSL, 1997).

⁸⁶ Structured Query Language.

⁸⁷ Vgl. Reibold, H. (Stilsprache für SGML, 2001).

⁸⁸ Vgl. Prescod, P. (Introduction to DSSSL, 1997).

⁸⁹ Vgl. Hofmann, T./Raitelhuber, U. (SGML/XML, 1998).

2.6. SGML-Werkzeuge

ment eingesetzt werden. Wird eine eigene DTD verwendet, muss der Editor gewöhnlich an diese angepasst werden.⁹⁰

Es gibt auch andere Möglichkeiten, SGML-Dokumente zu erstellen bzw. zu bearbeiten. So kann z.B. auch ein einfacher Texteditor oder ein Textverarbeitungsprogramm wie Microsoft Word genutzt werden. Diese Möglichkeiten führen jedoch nicht zu einem optimalen Ergebnis. Bei der Nutzung eines Texteditors müssen die Auszeichnungsbefehle manuell eingegeben werden – eine Vorgehensweise, die ein hohes Fehlerpotential aufweist. Bei Textverarbeitungsprogrammen ist die manuelle Befehlseingabe nicht nötig; hier wird die Formatierung über die individuellen Funktionen des Programms eingestellt. Allerdings muss das Dokument mit Hilfe eines Konvertierungsfilters in ein SGML-Dokument exportiert werden, was je nach Filter zu einem fehlerbehafteten Ergebnis führt.⁹¹

Ein auf SGML basierender Text wird in der Regel von einem sogenannten „Parser“-Programm gelesen.⁹² Mit Hilfe dieses Programms kann die Auszeichnung in SGML-konformen Dokumenten interpretiert werden.⁹³ Es überprüft zunächst die syntaktischen Strukturen des Dokuments und bestimmt, ob ein Zeichen eine Information oder einen Markierungsbefehl darstellt.⁹⁴ Weiterhin kann das Dokument auf Fehler untersucht werden; in der Regel wird geprüft, ob die Dokumenteninstanz mit der DTD übereinstimmt und ob diese der „SGML Declaration“ entspricht.⁹⁵ Weiterhin haben sie die Aufgabe, das Dokument in eine andere Form zu übertragen, die z.B. einer anderen Syntax unterliegt.⁹⁶

Schließlich ist der SGML-Browser zu nennen. Dieses Programm stellt das Dokument formatiert auf dem Bildschirm dar. Dabei wird jedem Element ein bestimmtes Layout oder eine bestimmte Funktion zugewiesen.⁹⁷

⁹⁰ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 248f.

⁹¹ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 47f.

⁹² Vgl. Reibold, H. (SGML – Ausgezeichnet, 2001).

⁹³ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 277.

⁹⁴ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 134.

⁹⁵ Vgl. van Herwijnen, E. (Practical SGML, 1994), S. 28.

⁹⁶ Vgl. Reibold, H. (SGML – Ausgezeichnet, 2001).

⁹⁷ Vgl. Hofmann, T./Raitelhuber, U. (SGML/XML, 1998).

3. Wirtschaftliche Bedeutung von SGML

Der SGML-Standard hat mittlerweile aufgrund seiner großen Vorteile im Zusammenhang mit der Datengenerierung, -haltung und -bearbeitung eine große wirtschaftliche Bedeutung erlangt. Viele angesehenen Unternehmen wie beispielsweise die Lufthansa, BMW⁹⁸ und IBM⁹⁹ haben sich diesem Standard verschrieben, um ihre Geschäftsprozesse möglichst effizient zu gestalten.¹⁰⁰

Nachfolgend sollen zunächst die allgemeinen wirtschaftlichen Vorteile von SGML erläutert werden. Anschließend werden diese Vorteile an den Beispielen Verlagswesen und Industrie verdeutlicht. Abschließend wird auf die Bedeutung von SGML für den weltweiten Datenaustausch eingegangen.

3.1. Wirtschaftliche Vorteile

Gemäß Arbortext lassen sich die wirtschaftlichen Vorteile auf acht wesentliche Punkte subsumieren: die Steigerung der Produktivität im Unternehmen, die Wiederverwendbarkeit der Daten, ihre Langlebigkeit, eine verbesserte Datenintegrität, eine bessere Datenkontrolle, eine höhere Datenteilbarkeit, die Ermöglichung eines problemlosen Datenaustausches und eine hohe Flexibilität bei der Datenausgabe.¹⁰¹

SGML bietet einen strukturierten Ansatz zur Generierung von Dokumenten aller Art. Die Autoren solcher Dokumente werden dazu angehalten, die Daten bzw. Informationen in vorgegebener Form zu organisieren. Dabei wird der Inhalt der Dokumente strikt von dessen Präsentation getrennt. Es ist dadurch möglich, die Aufbereitung der Daten zentral und im Sinne vorgegebener Richtlinien der Corporate Identity vorzunehmen. Die Autoren dieser Dokumente können sich so ausschließlich auf ihre ei-

⁹⁸ Bayerische Motorenwerke.

⁹⁹ Bei IBM ist heute schon XML im Einsatz; vgl. Pramberger, J. (XML hat bei IBM Geschichte und Zukunft, 1999); S. 53.

¹⁰⁰ Vgl. Thomas, P./Zimmermann, E. (SGML / ISO-Standard für Dokumentenverarbeitung, 1993).

¹⁰¹ Vgl. Arbortext (Getting started with SGML, 1992/1995).

3.1. Wirtschaftliche Vorteile

gentliche Aufgabe – die Generierung von Informationen – konzentrieren. Dies allein kann bereits ihre Produktivität in etwa verdoppeln.¹⁰²

Wiederverwendbarkeit der Daten bedeutet, dass auf SGML basierende Daten für die Generierung beliebig vieler und verschiedener Dokumente genutzt werden können. So kann jede Benutzergruppe die für sie relevanten Informationen bzw. Dokumentenarten aus nur einer gemeinsamen Datenbasis generieren.¹⁰³

Einer der wahrscheinlich größten Vorteile des SGML-Standards ist die Unabhängigkeit einer auf diesem Standard erzeugten Datenbasis von bestimmten Soft- und Hardwarestandards. Somit entfällt beim Austausch bestehender Soft- und Hardware-systeme durch entsprechend neuere und leistungsfähigere Systeme die lästige und fehleranfällige Konvertierung des bisherigen Datenbestandes. Das Unternehmen kann zudem jederzeit ohne Einschränkung auf die bereits vorhandenen Daten zurückgreifen. Durch diese Art der Datenlanglebigkeit und -integrität sollte sich die Handlungsfähigkeit eines Unternehmens deutlich erhöhen und dessen Umstellungskosten merklich sinken. Zudem wird durch diese Unabhängigkeit der Datenbasis auch ein problemloser Datenaustausch zwischen den verschiedenen Unternehmen eines Industriezweiges und Forschungseinrichtungen gewährleistet. Für den Erfolg dieser „externen“ Kommunikation bzw. entsprechender Netzwerke ist es essentiell, dass jeder Beteiligte Zugang zu jeder Information hat – quasi eine „Demokratisierung des Wissens“.¹⁰⁴ In diesem Kontext wird auch eine weitere Stärke von SGML sichtbar: die hohe Flexibilität bei der Datenausgabe. Die Daten von SGML-Dokumenten können problemlos auf den unterschiedlichsten Ausgabemedien dargestellt werden.¹⁰⁵ Dies ist insbesondere in einer Zeit, die durch große Fortschritte in der Informations- und Kommunikationstechnologie gekennzeichnet ist, von Bedeutung. Gewiss ist, dass die meisten Daten und Informationen, die bereits heute generiert werden, auch noch in einigen Jahren zur Verfügung stehen müssen. Ungewiss

¹⁰² Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹⁰³ Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹⁰⁴ Vgl. Plattner, H. in: Schneider, D./Schnetkamp, G. (E-Markets, 2000), S. 9.

¹⁰⁵ Vgl. Whatis?com (Stichwort SGML) sowie Meyer, M. (XML – was ist es?).

3.1. Wirtschaftliche Vorteile

ist hingegen, auf welchen Ausgabemedien die Ausgabe der Daten und Informationen dann gewünscht wird.¹⁰⁶

Ein weiterer großer Vorteil von SGML ist die bessere Datenkontrolle. SGML ermöglicht die Erstellung und Veränderung von „Informationselementen“ auf jeder Detailstufe. Die Identifizierung der entsprechenden Informationselemente wird dabei über das oder die zugewiesenen Attribute ermöglicht. Da Attribute „maschinenlesbar“ sind, können diese beispielsweise zur Verknüpfung zusammenhängender Informationen oder zur Organisation des Informationsflusses im Unternehmen genutzt werden. So kann definiert werden, welche Personen innerhalb eines Unternehmens berechtigt sind, bestimmte Informationen zu sehen bzw. zu editieren. Diese Automatisierung des Informationsflusses ermöglicht eine hohe Datenaktualität. Änderungen in der gemeinsamen Datenbasis des Unternehmens werden selbständig von allen Anwendungen, die darauf zurückgreifen, übernommen.¹⁰⁷

Als ebenfalls sehr vorteilhaft hat sich herausgestellt, dass SGML mit strukturierten Dokumentenkomponenten arbeitet. Diese Datenteilbarkeit ermöglicht es, neue Dokumente aus Datenbeständen unterschiedlicher Bereiche eines Unternehmens zusammenzusetzen. Verschiedene Benutzergruppen können somit die neuesten Informationen miteinander teilen, ohne dass eine Duplikation dieser Informationen notwendig wird. Ein Beispiel hierfür wären bestimmte Rechtsklauseln, die auf jedem Dokument und jeder Publikation des Unternehmens erscheinen sollen. Die Rechtsabteilung könnte die Aufgabe übernehmen, diese Informationen zu pflegen und den anderen Abteilungen zur Verfügung stellen. Ein einziger „tag“ in den Dokumenten und Publikationen der übrigen Abteilungen würde dann sicherstellen, dass immer auf die aktuellste Version dieser Rechtsklauseln zurückgegriffen wird.¹⁰⁸

¹⁰⁶ Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹⁰⁷ Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹⁰⁸ Vgl. Arbortext (Getting started with SGML, 1992/1995).

3.2. Anwendung im Verlagswesen

Insgesamt lässt sich festhalten, dass SGML insbesondere für Unternehmen geeignet ist, für deren Daten bzw. Informationen eins oder mehrere der folgenden Kriterien zutreffen sollen:¹⁰⁹

- Langlebigkeit, z. B. 20 Jahre.
- Kostengünstige Aktualisierungsmöglichkeit.
- Datenaustausch zwischen heterogenen Hard- und Softwareumgebungen.
- Verständliche Strukturierung auch längerer Dokumente.
- Dokumente beinhalten Daten bzw. Informationen anderer Abteilungen bzw. solche, die mehrere Abteilungen nutzen.
- Die Daten bzw. Informationen werden für die Ausgabe in verschiedenen Dokumenten benötigt.
- Die Daten bzw. Informationen ändern sich häufig und werden auch häufig benötigt.
- Die Daten bzw. Informationen müssen gemäß bestimmter Richtlinien wiedergegeben werden.

3.2. Anwendung im Verlagswesen

Das größte Anwendungsgebiet für SGML ist das Verlagswesen. Hier war es von Anfang an wichtig, Layout und Inhalt zu trennen.¹¹⁰

SGML erweist sich insbesondere bei der Produktion von elektronischen Dokumenten als sinnvoll, da sich hier die Inhalte dieser Dokumente oftmals überschneiden und eine „intelligente Verwaltung“ von Dokumenten und deren Inhalten gefordert wird. SGML erleichtert Autoren die Suche innerhalb der Dokumente, ermöglicht eine Verknüpfung zu anderen relevanten Inhalten und sorgt dafür, dass sich der gesamte Dokumentenbestand immer auf dem aktuellsten Stand befindet. Entsprechende Updates hierfür können aufgrund des hohen Datenvolumens nicht mehr manuell erfolgen, ein

¹⁰⁹ Vgl. Arbortext (Getting started with SGML, 1992/1995).

¹¹⁰ Vgl. Ittner, A. (XML-Einführung, 1999), S. 9.

3.3. Anwendung in der Industrie

automatischer Prozess ist erforderlich. Hierfür sind folgende Voraussetzungen nötig.¹¹¹

- die Trennung der Datenhaltung und –pflege von den Endanwendungen,
- die Haltung der Daten in neutraler Form,
- die Auszeichnung der Inhalte, nicht aber des Layouts, sowie
- die Generierung verschiedener Dokumente aus einer Quelle.

Diesen Voraussetzungen wird SGML gerecht. Hierdurch ist es den Verlagen möglich, Produkte in kürzester Zeit zu relaunchen, ihre Bestände automatisiert zu pflegen und redundante Datenhaltung zu vermeiden. SGML ermöglicht den Verlagen zudem eine formatgerechte Veröffentlichung des Inhalts in nahezu jeder Umgebung, unabhängig von bestimmter Soft- und Hardware.

3.3. Anwendung in der Industrie

In der Industrie erweist sich der Einsatz von SGML ebenfalls als vorteilhaft. Ziel ist es, die Papierflut im Unternehmen und die doppelte Erfassung von Daten zu verhindern oder zumindest zu minimieren. Handbücher, Nachschlagwerke und Formulare können auf SGML umgestellt werden. Letzten Endes stehen dieselben Gründe und Performancevorteile hinter diesem Schritt, wie sie im Verlagswesen vorliegen. Man ist bestrebt, redundanzfreie Daten kostengünstig und einheitlich immer auf dem aktuellsten Stand zu halten. Will man Inhalte erneuern, so sollte dies zeitnah und in einer zentralen Datenbasis geschehen, auf die sämtliche anderen Anwendungen zurückgreifen; das Layout muss sich dabei automatisch anpassen bzw. schon vorab definiert sein, um die Geschäftsprozesse effizient zu gestalten. Schließlich können diese Unternehmen signifikant von der Möglichkeit der Datenteilbarkeit profitieren.

¹¹¹ Vgl. Hofmann, T./Raitelhuber, U. (SGML/XML, 1998), S. 4.

3.4. Weltweiter Datenaustausch

Über die eigentlichen Auswirkungen von SGML auf Wirtschaft und Gesellschaft liegen keine konkreten Daten vor. Der Erfolg des Internets mit der auf SGML basierenden Sprache HTML ermöglicht aber schon bereits heute den Schluss, dass SGML in den vergangenen Jahren wesentlich an einer der wohl folgenschwersten wirtschaftlichen und gesellschaftlichen Umwälzungen beteiligt war.¹¹² Jeder Internetnutzer kennt HTML; dass aber im Endeffekt der Standard SGML dahintersteht, wissen die wenigsten.

Durch die Entwicklung der digitalen Technologien ist die qualitative Barriere zwischen Standards zur Übertragung von Informationen und Standards für den zu übertragenden Inhalt überwunden worden.¹¹³ So aggregiert HTTP¹¹⁴ TCP/IP¹¹⁵-„Informationspakete“ in Dokumente. Diese HTTP-Dokumente werden anschließend von HTML in Web-Seiten formatiert. Schließlich wandelt SGML oder XML diese Dokumente in ein breiteres Set selbstbeschreibender Dateien. Verschiedene Inhaltsstandards wie OFX¹¹⁶ oder OASIS¹¹⁷ setzen schließlich SGML bzw. XML zur Protokollierung entsprechender Informationen ein.¹¹⁸

Trotzdem sind bis heute fehlende (Inhalt-)Standards der Grund, dass Kunden und Lieferanten nicht miteinander kommunizieren können. Weltweit funktionieren e-Procurement-Systeme – egal ob bei Konzernen oder Marktplätzen – darum nicht wie gewünscht. Die technische Voraussetzung besteht zwar durch Sprachen wie SGML und XML; die inhaltliche Übersetzung der Informationen fehlt jedoch bis heute. Dieses Problem soll zukünftig beispielsweise durch Lösungen auf Basis von

¹¹² Vgl. Oetinger, B. von in: Evans, P./Wurster, T. S. (Web Attack, 2000), S.7; Plattner, H. in: Schneider, D./Schnetkamp, G. (E-Markets, 2000), S. 9.

¹¹³ Vgl. Evans, P./Wurster, T. S. (Web Attack, 2000), S.39f.

¹¹⁴ HyperText Transfer Protocol.

¹¹⁵ Transfer Control Protocol/Internet Protocol.

¹¹⁶ Open Financial Exchange.

¹¹⁷ Open Access Same-time Information System.

¹¹⁸ Vgl. Evans, P./Wurster, T. S. (Web Attack, 2000), S.39f.

3.4. Weltweiter Datenaustausch

eCl@ss, dem internationalen Standard für Produktdatenbeschreibung, gelöst werden.¹¹⁹

4. SGML – die Mutter aller Sprachen

SGML hat das Erscheinungsbild der heutigen Meta- und Markupsprachen stark geprägt. Als Metasprache beinhaltet SGML Vorschriften, die für die formale Definition weiterer Auszeichnungssprachen genutzt werden können. Davon wurde insbesondere bei den Sprachen HTML und XML Gebrauch gemacht. Folgende Abbildung verdeutlicht die Entwicklung dieser Sprachen:

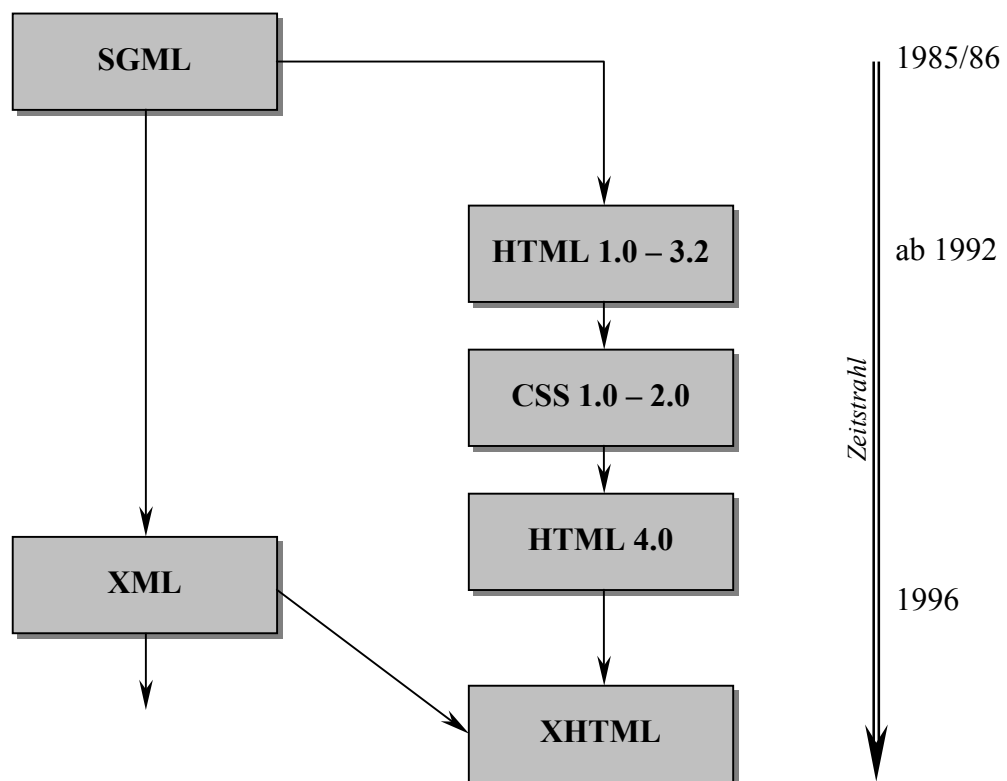


Abbildung 1: Entwicklungshistorie von SGML-basierten Sprachen¹²⁰

¹¹⁹ Vgl. tradecosmos (eCl@ss, 2001).

¹²⁰ Eigene erweiterte Darstellung in Anlehnung an Meyer, M. (XML – Was ist es?, o.J.); Unterholzer, G. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001).

4.1. HTML

4.1. HTML

Die erste, stark vereinfachte Anwendung von SGML war die HyperText Markup Language, kurz HTML. Ursprünglich wurde sie als reine Dokumentbeschreibungssprache für elektronisch zu übertragende Dokumente konzipiert; später wurde sie zur Erstellung von Internetseiten genutzt.¹²¹ Aufgrund des einfachen Aufbaus und der Tatsache, dass man HTML mit jedem Texteditor bearbeiten kann, hat sich diese Sprache zur Standardbeschreibungssprache des Internets entwickelt.¹²²

HTML wird durch eine SGML-DTD definiert, die in drei leicht unterschiedlichen Varianten vorliegt.¹²³ Diese bestimmen, dass ein HTML-Dokument aus einem Kopfbereich (head) und einem Rumpf (body) besteht. Der Kopf enthält unter anderem den Titel des Dokumentes, während der Rumpf den formatierten Inhalt darstellt¹²⁴. Darüber hinaus gibt die DTD die benutzbaren „tags“ und Attribute fest vor.

Dieser vereinfachte und leicht zu handhabende Standard wird gleichzeitig als Nachteil von HTML gesehen. Durch die Vorgabe der Markup-Befehle kann die Sprache nicht durch individuelle Definitionen erweitert werden.¹²⁵ Aus diesem Grund ist HTML für die Publikation auf unterschiedlichen Medien ungeeignet.¹²⁶ Im Laufe der Entwicklung der einzelnen HTML-Versionen konzentrierte man sich außerdem immer stärker auf die Kontrolle des Layouts anstatt – entsprechend dem Grundgedanken zu SGML – auf die Struktur. Mit dem reinen HTML-Standard ist daher eine Trennung von Inhalt und Format kaum möglich.¹²⁷

Um den zuletzt genannten Mangel zu beheben, haben Programmierer für HTML 3.0 eine Stilsprache namens Cascading Style Sheets (CSS) entwickelt. Erst durch ihre Nutzung ist es möglich, den Inhalt vom Layout zu trennen. CSS legt Rahmenbedin-

¹²¹ Vgl. Meyer, M. (XML – Was ist es?, o.J.).

¹²² Vgl. Macherius, J. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996).

¹²³ Vgl. Münz, S. (Selfhtml, 1998).

¹²⁴ Vgl. Ittner, A. (XML-Einführung, 1999), S. 13.

¹²⁵ Vgl. Meyer, M. (XML – Was ist es?, o.J.).

¹²⁶ Vgl. Macherius, J. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996).

¹²⁷ Vgl. Unterholzner, G. (HTML – Die Haus- und Hof-Sprache des Web, 2001).

4.2. Extensible Markup Language – XML

gungen fest, wie Auszeichnungen vom Browser interpretiert und dargestellt werden müssen¹²⁸. Mit ihrer Hilfe kann z.B. die Schriftart und –farbe für bestimmte Abschnitte des Dokuments festgelegt werden. Sie ist allerdings nicht so mächtig wie DSSSL, ihrem Analogon in SGML, ihre Funktionsvielfalt ist geringer. Beispielsweise können Texteinheiten nicht vorgegeben werden und an beliebigen Stellen im Dokument dargestellt werden. Auch die Verwaltung von Variablen ist nicht möglich. Eigenständiger Text, wie z.B. Seitenzahlen, lässt sich ebenfalls nicht realisieren.¹²⁹

4.2. Extensible Markup Language – XML

Angesichts der Mängel von HTML und der enormen Komplexität von SGML entstand 1996 eine weitere Sprache, die Extensible Markup Language (XML). Sie ist wie SGML eine Metasprache und ist als ihr direkter Nachfolger zu sehen. Allerdings ist XML hinsichtlich der Syntax wesentlich einfacher aufgebaut, da bei ihrer Erstellung komplexe und selten verwendete Eigenschaften aus SGML weggelassen wurden. XML ist somit eine „abgespeckte“ Version bzw. Teilmenge von SGML.¹³⁰ So formuliert Macherius treffend: „Wie radikal die Einschnitte sind, mag man dem Umstand entnehmen, dass die formale Definition von XML auf 33 Druckseiten möglich ist. SGML benötigt hierfür mehr als 500.“¹³¹

Der große Vorteil dieser Sprache ist schon aus dem Namen ersichtlich: „extensible“ bedeutet soviel wie erweiterbar. Da es sich bei XML ebenfalls um eine Metasprache handelt, könnte sich jeder Nutzer beliebig viele individuelle „tags“ definieren und sich so seine eigene Markup-Sprache schaffen¹³². Dies widerspräche jedoch der eigentlichen Aufgabe von XML, einen universellen Austausch von Dokumenten zu ermöglichen. Um einer Vielzahl von – aus XML entwickelten – Auszeichnungsspra-

¹²⁸ Vgl. Unterholzer, K. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001).

¹²⁹ Vgl. Macherius, J. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996).

¹³⁰ Vgl. Macherius, J. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996).

¹³¹ Macherius, J. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996).

¹³² Vgl. Unterholzer, G. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001).

4.2. Extensible Markup Language – XML

chen entgegenzuwirken, wurden nur bestimmte Dokumenttypen von der W3C offiziell anerkannt.¹³³

XML weist auch sonst große konzeptionelle Ähnlichkeiten zu ihrem Vorgänger SGML auf. Die generischen Informationen zu den verwendbaren Elementen von XML liegen in Form einer DTD vor. Weiterhin erfolgt eine eindeutige Trennung von Inhalt, Struktur und Layout. Die Struktur wird durch semantische „tags“ vorgegeben, während das Layout durch eine Stilsprache bestimmt wird. Aufgrund des Konzepts von XML ist die Nutzung einer Stilsprache unabdingbar, während ein CSS für die Darstellung eines HTML-Dokumentes nicht zwingend erforderlich ist.

Vergleicht man XML mit HTML, so kann man einen Vorteil klar herausstellen: die erweiterte Suchfunktion, die auch semantische Inhalte berücksichtigt. Diese lässt sich an folgendem Beispiel erklären, in welchem die Darstellung von Format und Inhalt mittels HTML und XML gegenübergestellt werden:

	HTML	XML
Quelltext	<pre><p>Hermann Hesse
Frankfurter Straße 8
69476 Wiesbaden
geb. 3. März 1927</p></pre>	<pre><person id= "p1876"> <autor> <vorname>Hermann</vorname> <nachname>Hesse</nachname> </autor> <adresse><strasse> Frankfurter Straße 8 </strasse><ort>69478 Wiesba- den</ort></adresse> <geburtstag><tag>3</tag> <monat>März</monat> <jahr>1927</jahr> </geburtstag></person></pre>
Darstellung im Browser	Hermann Hesse Frankfurter Straße 8 69476 Wiesbaden geb. 3. März 1927 <i>(Exakte Formatierung abhängig von der jeweils gewählten Stilsprache)</i>	Hermann Hesse Frankfurter Straße 8 69476 Wiesbaden geb. 3. März 1927 <i>(Exakte Formatierung abhängig von der jeweils gewählten Stilsprache)</i>
Tags	Es wird nur das Format des Textes durch formatorientierte „tags“ beschrieben.	Es werden auch semantische „tags“ zur Beschreibung des Inhalts gesetzt.

Abbildung 2: Gegenüberstellung von HTML und XML¹³⁴

¹³³ Vgl. Unterholzer, G. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001).

¹³⁴ Eigene Darstellung in Anlehnung an Macherius, I. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996); Unterholzer, G. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001).

4.2. Extensible Markup Language – XML

Der Vorteil dieser Art des „tagging“ liegt auf der Hand. Man kann beispielsweise gezielt nach dem Nachnamen suchen, da die semantischen „tags“ eine Selektion stark vereinfachen. Sucht man im Internet z.B. nach dem Autor „Hermann Hesse“, so erhält man bei auf HTML basierenden Dokumenten alle Begriffe, in denen „Hesse“ vorkommt, auch wenn sie vom Kontext mit dem gesuchten Autor nichts zu tun haben. Basiert das Dokument aber auf einem XML-Standard, so ist eine „intelligente Suche“ möglich. Der Rechner überprüft auch die semantischen „tags“, d.h. er bildet nur solche Begriffe ab, die das Wort „Hesse“ enthalten und die durch einen „tag“ inhaltlich als „Autor“ ausgewiesen werden.¹³⁵

Des Weiteren wird mit XML das Problem der unterschiedlichen Darstellung in verschiedenen Browsern gelöst. HTML weist hier Schwachstellen auf. Eine direkt aus XML resultierende Markup-Sprache könnte HTML deshalb in Zukunft ablösen.

Für XML wurde – nach einem Vorschlag von Microsoft, Inso und Arbortext – eine eigene Stilsprache entwickelt, die sich stärker an DSSSL orientiert: die Extensible Stylesheet Language oder kurz XSL. Mit Hilfe von XSL können Dokumente verschiedentlich darstellen werden, je nach Bedarf des Benutzers. Zudem unterstützt es auch unterschiedliche Ausgabemedien.¹³⁶ Im Grunde ließe sich auch das für HTML genutzte CSS verwenden. Dies ist allerdings mit Nachteilen verbunden, da sich CSS lediglich auf die Darstellung des Markups beschränkt, jedoch keine programmtechnischen Konstrukte ermöglicht.¹³⁷ Momentan kommt XSL insbesondere bei der Umwandlung von XML-Dokumenten in HTML-Dokumente zum Einsatz.

¹³⁵ Vgl. u.a. Dünhölter, K. (Das Web automatisieren mit XML, 1998).

¹³⁶ Vgl. Martiné, A. (XSL – Voll im Bilde, 2001).

¹³⁷ Martiné, A. (XSL – Voll im Bilde, 2001).

4.2. Extensible Markup Language – XML

Zu den anerkannten Markup-Sprachen, die sich aus XML entwickelt haben, zählen unter anderem folgende Sprachen:

Sprache	Erklärung
XHTML ¹³⁸	<ul style="list-style-type: none"> • Dokumententyp von XML • HTML 4.0 wurde in Einklang mit XML gebracht, d.h. alle Elemente von HTML 4.0 sind in XHTML so formuliert, dass sie den Vorgaben und Anforderungen von XML entsprechen
SVG ¹³⁹	<ul style="list-style-type: none"> • beschreibt zweidimensionale Grafiken (Vektorgrafiken, Bilder, grafischen Text), die ohne Plugin dargestellt werden können • ermöglicht Verkleinerungen und Vergrößerungen ohne Qualitätsverlust (True Color) • Volle Integration der Grafiken in das HTML-Format
VoiceXML	<ul style="list-style-type: none"> • Ein Dokument wird über diese Sprache sowohl als Internetfile als auch als automatische Telefonansage veröffentlicht • Programmierung eines MP3¹⁴⁰ oder eines anderen Audio-Files entfällt
SyncML ¹⁴¹	<ul style="list-style-type: none"> • dient dem Abgleich von Daten aus unterschiedlichen Plattformen • kann zwischen Unternehmensdatenbanken, web-basierten Dokumenten, E-Mails, etc. einen redundanzfreien Datenaustausch ausführen
WML ¹⁴²	<ul style="list-style-type: none"> • Schaffung von Websites für das Handy • Ausgerichtet auf die niedrigen Bandbreiten von PDAs¹⁴³, Pagem und Telefonen • Einfache Formatierungen und Grafiken möglich
CML ¹⁴⁴	<ul style="list-style-type: none"> • Erweiterung für die Darstellung chemischer Molekülstrukturen
CDF ¹⁴⁵	<ul style="list-style-type: none"> • ermöglicht die automatische Belieferung mit selbst zusammengestellten Webinhalten • Grundlage der Push-Technologie im Internet

Abbildung 3: XML-basierte Markup-Sprachen¹⁴⁶

¹³⁸ Extended Hypertext Markup Language.

¹³⁹ Scalable Vector Graphics.

¹⁴⁰ MPEG 2.5 Audio Layer III.

¹⁴¹ Synchronisation Markup Language.

¹⁴² Wireless Markup Language.

¹⁴³ Personal Digital Assistants.

¹⁴⁴ Chemical Markup Language.

¹⁴⁵ Channel Definition Format.

¹⁴⁶ Eigene Darstellung in Anlehnung an Unterholzer, K. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001).

Folgende Abbildung soll den Zusammenhang von SGML und XML sowie ihrer Stilsprachen und aus ihnen entwickelten Markup-Sprachen noch einmal zusammenfassen. Rechtecke mit den runden Kanten stellen dabei die Metasprachen dar, die normalen Rechtecke zeigen Markup-Sprachen und die Kreise stehen für Stilsprachen:

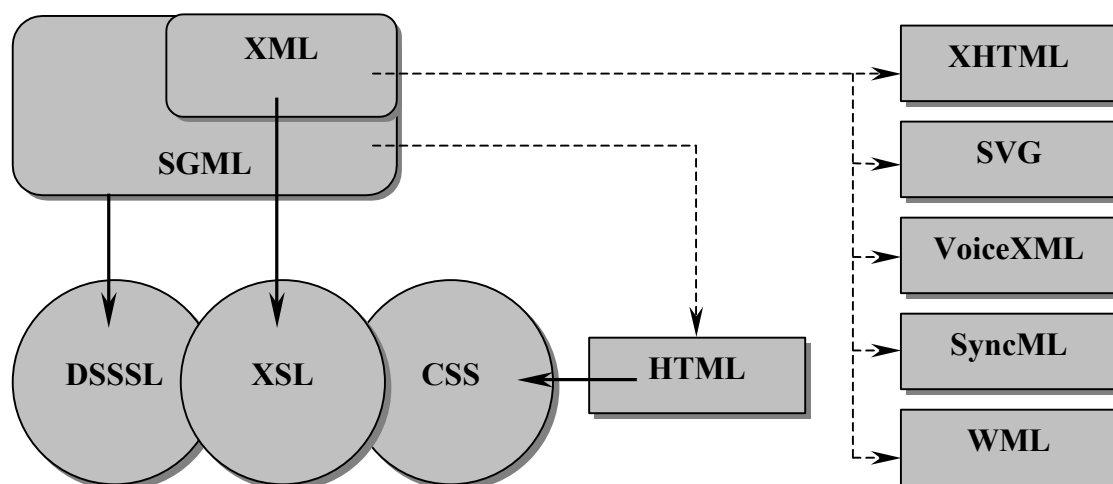


Abbildung 4: Zusammenhang zwischen SGML und XML¹⁴⁷

5. Fazit und Ausblick für die Zukunft

Ziel der vorliegenden Arbeit war es, dem Leser die Sprache SGML nahe zu bringen. Die Historie von SGML, das Grundkonzept und der Zusammenhang mit XML wurden verdeutlicht.

Die Grundidee – die Trennung von Struktur und Layout – führte zur Entwicklung von SGML mit allen daraus resultierenden Markup- und Stilsprachen. XML ist ein Produkt dieser Entwicklung, welches die Vorteile von SGML, aber nicht seine Nachteile beinhaltet. Aufgrund der geringeren Komplexität ist XML wesentlich einfacher zu verstehen und genießt daher – im Gegensatz zu SGML – nicht den Ruf

¹⁴⁷ Eigene Darstellung in Anlehnung an Unterholzer, G. (HTML, XML, SGML & Co. – Die Sprachen des Internets, 2001); Pramberger, J. (XML hat bei IBM Geschichte und Zukunft, 1999); S. 53.

einer „Geheimsprache“.¹⁴⁸ Dies ist ein weiterer Grund für die heutige Popularität dieser Sprache. Deshalb wird XML die Sprache sein, die den Grundgedanken von SGML verbreiten und den dadurch begonnen Trend fortsetzen wird. XML weist das Potenzial auf, um möglicherweise HTML in der Zukunft abzulösen.

Die Bereitstellung von Inhalt, Struktur und Layout wird in naher Zukunft nicht nur hard- und softwareunabhängig sein. Der Trend geht unseres Erachtens dahin, dass mit Hilfe der Metasprachen SGML und XML diese drei Elemente eines Dokumentes benutzer- oder benutzergruppenspezifisch dargestellt werden. Man wird versuchen, die Darstellungsform an den Bedürfnissen dieser Gruppen auszurichten und Ihnen das Dokument entsprechend zu präsentieren.

In einem Unternehmen wird die vollständige Implementierung dazu führen, dass verschiedene Abteilungen dieselben Informationen unterschiedlich präsentiert bekommen. So wird es z.B. bei der Vorstellung eines neuen Autos dazu kommen, dass der Verkaufsabteilung eher die Vertriebsaspekte präsentiert werden, während der Blick der Techniker auf die technischen Details fokussiert sein wird. Alle Beteiligten werden auf das gleiche Dokument zugreifen können, nur die Ausgabe wird differieren.

Metasprachen dieser Art haben die Welt der Informatik bereits verändert, aber die Entwicklung geht weiter. Letztendlich wird jeder Nutzer von elektronischen Informationen über einen genau auf seine Bedürfnisse zugeschnitten Arbeitsplatz verfügen. Das Internet wird sich ihm ebenfalls ganz individuell darstellen. Was diese Entwicklung für Marketing und Vertrieb bedeutet, ist noch nicht abzusehen.

Tatsache ist, dass SGML mit der Trennung von Inhalt und Struktur den ersten Grundstein für diese informationsorientierte Entwicklung gelegt hat. Die hinter dieser Sprache liegenden Kerngedanken sind – obwohl die Sprache selbst schon älter ist – aktueller denn je.

¹⁴⁸ Vgl. Macherius, I. (XML: Professionelle Alternative zu HTML – Revolution der Experten, 1996).

Literatur**Literaturquellen**

- Bryan, Martin** „SGML - An Author's Guide to the Standard Generalized Markup Language“, 1. Auflage, Verlag Addison-Wesley, Menlo Park (CA) et al., 1988.
- Evans, P./
Wurster, T.S.** „Web Attack“
München 2000
- Herwijnen, Eric van** „Practical SGML“,
2nd Edition“, Verlag Kluwer Academic
Publishers, Boston / Dordrecht / London, 1994
- Holloway, H. L.** „Generic Coding and SGML“
1987
- Ittner, Ariane** „XML-Einführung“
Viser Systemintegration, Stuttgart
erschienenen in: *Datenintegration durch XML – Chancen,
Anwendungen und Perspektiven im E-Business*
Viser Systemintegration, Stuttgart 1999
- Lobster Network Sto-
rage AG** „Data Storage“
September 2000
- Oetinger, B. von** Vorwort zu „Web Attack“
Herausgeber: Evans, P./ Wurster, T.S.
München 2000
- Plattner, H.** „E-Markets“
Herausgeber: Schneider, D./ Schnetkamp, G.
Wiesbaden 2000
- Pramberger, Johann** „XML hat bei IBM Geschichte und Zukunft“
IBM Deutschland Entwicklung GmbH
erschienenen in: *Datenintegration durch XML – Chancen,
Anwendungen und Perspektiven im E-Business*
Viser Systemintegration, Stuttgart 1999

Internetquellen

- Arbortext „Getting started with SGML“
1992/1995
<http://www.arbortext.com>
- Dünhölter, Kuno „Das Web automatisieren mit XML“
1998
<http://members.aol.com/xmldoku>
- Goldfarb, Charles F. „The Roots of SGML -- A Personal Recollection“
1996
<http://www.sgmlsource.com/history/roots.htm>
- Hofmann, Thomas /
Raithelhuber, Ursula „SGML/XML“
01.10.1998
<http://th-o.de/sgml/sgmlv.htm>
- Macherius, Ingo „XML: Professionelle Alternative zu HTML – Revolution
der Experten“
erschieden in *iX online* 1996
<http://www.heise.de/ix/1997/06/106/artikel.html>
- Martiné, Andrea „Die Grammatik des Webs“
2001
http://www.computerchannel.de/knowhow/dtd/dtd_1.phtml
- Meyer, Michael „XML – Was ist es?“
<http://home.t-online.de/home/meyer.michael/xml/main.htm>
- Münz, Stefan „Selfhtml“
1998
<http://www.teamone.de/selfhtml/tcab.htm>
- Prescod, Paul „Introduction to DSSSL“
July 1997
<http://www.prescod.net/dsssl>

Internetquellen

-
- Reibold, Holger „Stilsprache für SGML“
07.03.2001
http://www.computerchannel.de/knowhow/dsssl/dsssl_1.html
- Reibold, Holger „SGML – Ausgezeichnet“
2001
<http://www.computerchannel.de/artikel.phtml?id=2170>
- Ruettgers, Michael C. „E-Infostructure“
<http://www.emc2.ch/about/einfostructure>
- SGML User's Group „A Brief History of the Development of SGML“
11.06.1990
<http://www.sgmlsource.com/history/sgmlhist.htm>
- Sperberg-McQueen, C.M./ Burnard, Lou
Editors „A gentle Introduction to SGML“
<http://www-tei.uic.edu/orgs/tei/sgml/teip3sg/SG.htm>
- Thomas, Patrick/
Zimmermann, Edward „SGML / ISO-Standard für Dokumentenverarbeitung“
April 1993
http://dodo.bsn.com/BSN/BSn/Intros/SGML_gr.html
- tradecosmos GmbH „eCl@ss“
<http://www.tradecosmos.de/Produkte/eclass.htm>
- Unterholzer, Günter „HTML, XML, SGML & Co. – Die Sprachen des Internets“
09.03.2001
<http://www.computerchannel.de/artikel.phtml?id=2144>
- Whatis?com Stichwörter:
- DSSSL
- DTD
- GML
- SGML
<http://whatis.techtarget.com>