
Diplomarbeit im Studiengang Medieninformatik

**Evaluierung von Lösungswegen
zur cross-medialen Umsetzung
eines wissenschaftlichen Periodikums**

vorgelegt von Julia Schlatterer

an der Fachhochschule Stuttgart – Hochschule der Medien

am 1. August 2006

- 1. Prüfer: Prof. Dr.-Ing. Oliver Kretzschmar**
 - 2. Prüfer: Prof. Dr.-Ing. Michael M. Resch**
-

Hiermit versichere ich,
dass ich die vorliegende Arbeit ohne unerlaubte Hilfe
und ohne Benutzung anderer als der angegebenen Hilfsmittel
angefertigt habe.

Alle Stellen, die wörtlich oder sinngemäß
aus Veröffentlichungen entnommen wurden,
sind als solche kenntlich gemacht.

1. August 2006

Abstract

In der vorliegenden Arbeit werden verschiedene Lösungswege zur cross-medialen Umsetzung eines wissenschaftlichen Periodikums evaluiert. Die Anforderungen werden anhand eines konkreten Anwendungsbeispiels – der Fachzeitschrift ‚inSiDE‘ – formuliert. Da dieses Anwendungsbeispiel sowohl als Printausgabe als auch Online-Medium erscheint, muss der Web Workflow in den bereits vorhandenen Print Workflow integriert werden, um einen gemeinsamen Publikationsprozess zu schaffen. Der besondere Evaluationsbedarf besteht in der Erstellung, Verarbeitung und Darstellung mathematischer Ausdrücke in einem Publikationsprozess wissenschaftlicher Texte. Dieser Problematik wird folgerichtig besondere Aufmerksamkeit gewidmet.

Inhaltsverzeichnis

Abbildungsverzeichnis	VI
Tabellenverzeichnis	VIII
Abkürzungsverzeichnis	VIII
1 Einleitung	1
2 Problemstellung	4
2.1 Anwendungsbeispiel: die Fachzeitschrift ‚inSiDE‘	5
2.2 Ist-Zustand eines beispielhaften Publikations-Workflow	5
2.3 Zustandsanalyse	8
2.4 Ziel und Aufgabenstellung	11
2.5 Anforderungsprofil	12
2.5.1 Anspruch wissenschaftlicher Texte	12
2.5.2 Anforderungen	13
3 Evaluation Teil I – Vorstellung der Evaluationsobjekte	17
3.1 Lösungsmöglichkeiten der Inhaltsbeschaffung	20
3.1.1 Texterstellung in Word	21
3.1.2 Eingabe von Formeln in Word	21
3.1.3 Integration von Bildern in Word	23
3.1.4 Texterstellung und Integration von Bildern über eine Web-Maske	24
3.1.5 Eingabe von Formeln über eine Web-Maske	25
3.2 Lösungswege zur Erzeugung von XML	28
3.2.1 XML aus Word erzeugen	28
3.2.2 XML über eine Web-Maske erzeugen	29
3.2.3 XML aus einem Layout-Programm erzeugen	29
3.3 Integration von nicht text-basierten Inhalten in XML	30
3.3.1 Integration von Formeln in XML	30
3.3.2 Austausch von Bilddateien über die Integration in XML	31
3.4 Paralleler versus sequentieller Publikationsprozess	32
3.4.1 Cross-medialer Publikationsprozess – Single-Source Publishing	33
3.4.2 Sequentiell verlaufender Publikationsprozess – Erst Print dann Web	36
3.5 Content Management Systeme als integrativer Lösungsansatz	40
3.6 Bestimmung des Kriterienkataloges	41

4	Evaluation Teil II – Bewertung und Ergebnis	48
4.1	Beurteilung der einzelnen Lösungsmodule	48
4.2	Überführung der Merkmalsausprägungen in Nutzwerte	55
4.3	Gewichtung der Bewertungskriterien	58
4.4	Interpretation der Bewertungsergebnisse	61
4.5	Ergebnis der Evaluation.....	64
5	Umsetzung der evaluativ abgeleiteten Handlungsempfehlung	67
5.1	Datenstruktur	67
5.2	Speicherung und Verwaltung von Daten	71
5.3	Darstellung der Daten	72
5.4	Automatisierung des Web Workflow	73
6	Fazit und Ausblick	76
	Literaturverzeichnis	X
	Elektronisches Quellenverzeichnis	XI

Abbildungsverzeichnis

Abb. 1: Aufbau und Struktur der vorliegenden Arbeit	2
Abb. 2: Schematische Darstellung eines beispielhaften Ist-Zustandes zwei verschiedener Produktionsabläufe	6
Abb. 3: Schematische Darstellung der Inhaltsbeschaffung	18
Abb. 4: Schematische Darstellung des Datenaustausch und des Prozessablaufes von Print und Web	19
Abb. 5: Parallel versus sequentiell ablaufender Workflow-Prozess	20
Abb. 6: Screenshot der Zeichentabelle der Funktion ‚Symbol einfügen‘ in Word	22
Abb. 7: Screenshot der Symbolleiste des MS Formeleditor	22
Abb. 8: Screenshot der Anwendung MathType	23
Abb. 9: Beispiel: Java-Applet ‚WebEQ Input Control‘ als Browser-integrierte Symbolleiste zur Formeleingabe	27
Abb. 10: Mathematisches Formelwerkzeug als JavaScript-Anwendung zur Anzeige der Unicode-Adresse von Formeln, Beispiel	27
Abb. 11: Schematische Darstellung der Integrationsmöglichkeit von Artikelelemente wie Formeln und Bilder in die XML-Auszeichnung	31
Abb.12: Ablaufreihenfolge schematisch dargestellt: Parallel und sequentiell ablaufenden Lösungswege	32
Abb. 13: Schematische Darstellung des parallelen Workflow	34
Abb. 14: Schematische Darstellung des sequentiellen Workflow	37
Abb. 15: Lösungsmodul ‚Formelerstellung‘	41
Abb. 16: Lösungsmodul ‚Bildintegration‘	43
Abb. 17: Lösungsmodul ‚XML-Erzeugung‘	44
Abb. 18: Lösungsmodul ‚Kosten‘	45
Abb. 19: Lösungsmodul ‚Prozessablauf‘	46
Abb. 20: Die drei Lösungswege, kombiniert aus den jeweils bestmöglichen Elementen der entsprechenden Module	64
Abb. 21: Auszug des DTD-Entwurfs der Fachzeitschrift ‚inSiDE‘	70
Abb. 22: Screenshot eines Print Stylesheet in Form von Absatzformaten	72
Abb. 23: Ausschnitt aus einem Web Stylesheet in CSS-Form	73
Abb. 24: Screenshot der Online-Ausgabe ‚inSiDE‘, http://inside.hlrs.de	74

Tabellenverzeichnis

Tabelle 1:	Vor- und Nachteile der verschiedenen Optionen der Formelkodierung	26
Tabelle 2a:	Beurteilung der einzelnen Lösungsmodule: Formelerstellung in Word	49
Tabelle 2b:	Beurteilung der einzelnen Lösungsmodule: Formelerstellung über eine Web-Maske	50
Tabelle 3:	Beurteilung der einzelnen Lösungsmodule: Formelkodierung	51
Tabelle 4:	Beurteilung der einzelnen Lösungsmodule: Bildintegration	51
Tabelle 5:	Beurteilung der einzelnen Lösungsmodule: XML-Erzeugung	53
Tabelle 6:	Beurteilung der einzelnen Lösungsmodule: Kosten	53
Tabelle 7:	Beurteilung der einzelnen Lösungsmodule: Prozessablauf	54
Tabelle 8:	Überführung der Merkmalsausprägungen in Nutzwerte: Formelkodierung	56
Tabelle 9:	Überführung der Merkmalsausprägungen in Nutzwerte: Formelerstellung	56
Tabelle 10:	Überführung der Merkmalsausprägungen in Nutzwerte: Bildintegration	57
Tabelle 11:	Überführung der Merkmalsausprägungen in Nutzwerte: XML-Erzeugung	57
Tabelle 12:	Überführung der Merkmalsausprägungen in Nutzwerte: Kosten	57
Tabelle 13:	Überführung der Merkmalsausprägungen in Nutzwerte: Prozessablauf	58
Tabelle 14:	Gewichtung der Bewertungskriterien: Formelkodierung	59
Tabelle 15:	Gewichtung der Bewertungskriterien: Formelerstellung	60
Tabelle 16:	Gewichtung der Bewertungskriterien: Bildintegration	60
Tabelle 17:	Gewichtung der Bewertungskriterien: XML-Erzeugung	60
Tabelle 18:	Gewichtung der Bewertungskriterien: Kosten	61
Tabelle 19:	Gewichtung der Bewertungskriterien: Prozessablauf	61
Tabelle 20:	Punkteverteilung und Endergebnis	65

Abkürzungsverzeichnis

CSS	Cascading Stylesheet
DTD	Document Type Definition
DTP	Desktop Publishing
EPS	Encapsulated Post-Script
GIF	Graphic Interchange Format
HTML	Hypertext Markup Language
JPEG	Joint Photographic Expert Group
MathML	Mathematical Markup Language
PDF	Portable Document Format
PNG	Portable Network Graphics
SVG	Scalable Vector Grafics
TIFF	Tagged Image File Format
URL	Uniform Resource Locator
W3C	World Wide Web Consortium
XSL	Extensible Stylesheet Language
XSL-FO	Extensible Stylesheet Language-Formatting Objects
XSLT	Extensible Stylesheet Language Transformations
XML	Extensible Markup Language

1 Einleitung

„Die Medienindustrie wird durch die sich entwickelnden Bedürfnisse digitalen Publizierens mit einer großen Herausforderung für ihre Strukturen und Arbeitsweisen konfrontiert.“¹

Wie diese Aussage zeigt, ist digitales Publizieren mittlerweile kein Fremdbegriff mehr. Artikel, Diskussionsbeiträge, aber auch komplette Zeitungen und Zeitschriften werden vielfach durch elektronische Ausgaben bzw. Zusatzelemente ergänzt – oder teilweise sogar nur in dieser Publikationsform veröffentlicht. Für die Erzeugung des Publikationsproduktes ergibt sich hieraus jedoch ein Problem: Die beiden Ausprägungen der Publikationserstellung – traditionelles Publishing auf der einen, elektronisches auf der anderen Seite – werden im Allgemeinen als zwei voneinander unabhängige und unterschiedliche Produktionsprozesse angesehen. Das getrennte Publizieren von Print- und elektronischer Ausgabe führt daher in der Regel dazu, dass beträchtlicher Mehraufwand in der Publikationserstellung entsteht. Da aber prinzipiell beide Publikationsprodukte auf demselben Datenbestand basieren, würde eine Zusammenführung der beiden Prozesse zu einem erheblichen Effizienzgewinn in der Erstellung führen. Das Problem besteht also darin, einen Produktionsprozess zu finden, der die Erstellung von Print- und elektronischer Ausgabe miteinander vereint und auf einen Datenbestand zugreift. Diese Fragestellung soll im Folgenden näher untersucht werden.

Um aus einem vorhandenen Datenbestand verschiedene Medien zu generieren, ist es notwendig, Medienneutralität zu gewährleisten.² Der vom World Wide Web Consortium (W3C) empfohlene Standard für Texte beruht hierbei auf der medienneutralen Auszeichnungssprache Extensible Markup Language (XML)³. Zahlreiche Anwendungsprogramme bieten mittlerweile die entsprechenden Möglichkeiten an, um Dokumente in ein XML-Format zu konvertieren. Offen bleibt jedoch die Frage: Wie wird XML am sinnvollsten eingesetzt bzw. über welchen Weg kann XML gezielt in einen bestehenden Produktionsprozess integriert werden?

Des Weiteren beinhalten Publikationen meist nicht nur Texte, sondern auch Bilder und Grafiken, die nicht auf dem XML-Standard basieren bzw. nicht unmittelbar hierfür geeignet sind, aber ebenso medienneutral verwaltet werden müssen. Dabei kommt im Idealfall eine Mediendatenbank zum Einsatz, in der die Objekte strukturiert gespeichert und verwaltet werden können.

Einen kritischen Gesichtspunkt beim Publizieren wissenschaftlicher Texte stellen darüber hinaus mathematische Ausdrücke und Formeln dar. Dieser Aspekt lässt sich zwar durch die Verwendung

1 So Lionel Barber, Chefredakteur der Financial Times. Vgl. Spiegel Online (2006)

2 Unter einer medienneutralen Datenhaltung wird allgemein verstanden, dass Produkte, die in Datenformaten abgelegt werden, nicht nur auf eine Ausgabeform hin abgestimmt sein dürfen.

3 „XML ist ein universelles Format für strukturierte Daten. Während in einem HTML-Dokument, die Daten an ein bestimmtes Erscheinungsbild gebunden sind, ist dies bei XML nicht der Fall. XML-Tags beschreiben im Gegensatz zu HTML lediglich die Bedeutung des Inhalts, nicht seine Darstellung.“ Zitat aus Nix (2004)

der in einigen Textverarbeitungsprogrammen integrierten Formeleditoren zum Teil entschärfen. Was aber geschieht, wenn Texte inklusive Formeln verarbeitet und publiziert werden müssen?

In dieser Arbeit werden verschiedene Lösungswege zur cross-medialen⁴ Publikation eines wissenschaftlichen Periodikums vorgestellt. Schwerpunktmäßig geht es hierbei darum, den Web Workflow in den bereits vorhandenen Print Workflow zu integrieren und so einen gemeinsamen Publikationsprozess zu schaffen. Ein besonderes Augenmerk liegt dabei auf der Visualisierung mathematischer Ausdrücke, die in wissenschaftlichen Texten häufig vorkommen. Mathematische Formeln enthalten in der Regel griechische Buchstaben und Sonderzeichen, die im ASCII-Zeichensatz nicht enthalten sind. Somit stellt die Darstellung dieser Formeln einen besonderen Anspruch an die Arbeit dar. Davon ausgehend werden verschiedene Anwendungsmöglichkeiten erörtert, die unter den gegebenen Anforderungen durchführbar sind. Diese Anforderungen, die im Laufe der folgenden Evaluation der verschiedenen Lösungswege berücksichtigt werden sollen, werden am Beispiel der Fachzeitschrift ‚inSiDE‘ dargestellt.

Anhand der folgenden grafischen Übersicht (Abbildung 1) wird der Ablauf und Aufbau der Arbeit veranschaulicht.



Abb. 1: Aufbau und Struktur der vorliegenden Arbeit

⁴ Cross-medial bedeutet in diesem Zusammenhang medienübergreifendes Publizieren, wobei Inhalte mehrfach bewertet werden sollen. Vgl. Kretzschmar/Dreyer (2004), S. 189

Die inhaltliche Struktur der Arbeit bildet sich aus der Workflow-Gestaltung (Makro-Thema), die sich auf den Prozessablauf und die Integration des Web Workflow bezieht und der Formelverarbeitung (Mikro-Thema), die den besonderen Anspruch wissenschaftlicher Texte behandelt.

Wie in Abbildung 1 ersichtlich, wird in Kapitel 2 die Problemstellung dieser Arbeit ermittelt. Anhand einer beispielhaften Anwendung für wissenschaftliche Publikationen wird der Status quo eines Publikations-Workflow beschrieben und analysiert. Unter Berücksichtigung heutiger Kenntnisse soll aus dem weiteren Verlauf dieses Kapitels ein grundlegender Lösungsansatz zu der vorhandenen Problematik hergeleitet werden können.

Basierend auf einem Anforderungskatalog wird die Evaluation durchgeführt, die sich aus Gründen der Übersichtlichkeit und der Nachvollziehbarkeit in zwei Abschnitte teilt. In Kapitel 3 wird der erste Teil behandelt, in dem verschiedene Möglichkeiten von Lösungswegen als Evaluationsobjekte festgelegt werden. Dabei werden die Lösungswege in die Phasen der Artikelerstellung inkl. Formel- und Bildintegration (Inhaltsbeschaffung), dem Datenaustausch (Inhaltsübergabe) und der Produktveredelung⁵ (Inhaltsverarbeitung) unterteilt.

In Kapitel 4 werden die aufgezeigten Evaluationsobjekte mit Hilfe einer semi-quantitativen Nutzwertanalyse bewertet. Die auf Basis der Bewertung ermittelte beste Lösung ergibt sich aus den gewonnenen Ergebnissen; ihre Umsetzung im vorliegenden Fall wird deshalb empfohlen. Bewertung und Ergebnis bilden den zweiten Teil der Evaluation. Kapitel 5 bietet dann eine konzeptionelle Handlungsempfehlung zur Umsetzung des empfohlenen Lösungsweges. In Kapitel 6 wird abschließend eine Zusammenfassung der Stärken und Schwächen des gewählten Ansatzes und daraus sich möglicherweise als Folgewirkung ergebender Risiken und Chancen gegeben; zudem wird ein Ausblick auf mögliche Weiterentwicklungen des Publikations-Workflow am Höchstleistungsrechenzentrum Stuttgart bzw. eine mögliche Anpassung anderer Projekte an den vorgeschlagenen Lösungsweg gewagt.

Das Ziel dieser Arbeit ist die konzeptionelle Orientierung. Deshalb soll hier weitgehend auf technische Definitionen verzichtet werden, da diese den Rahmen der Arbeit sprengen würden und vielfach auch als bekannt angenommen werden können.

⁵ Unter Produktveredelung soll hier Folgendes verstanden werden: Die Veredelung von Inhalten ist ein Prozess, in dem die Inhalte zu hochwertigen Endprodukten aufbereitet werden. Dies kann rein optische Zwecke haben, aber auch die Produktivität steigern.

2 Problemstellung

In diesem Kapitel wird der typische Produktions-Workflow eines Druckproduktes, das zusätzlich im Web veröffentlicht wird, beschrieben und kritisch analysiert. Dabei wird auf die auftretenden Schwächen in diesem Workflow eingegangen. Anschließend werden hierzu Zielsetzungen aufgeführt und erläutert, die es zu erreichen gilt, um die bestehenden Probleme zu lösen bzw. zum Vorteil zu verändern.

Da es verschiedenartige, unterschiedlich häufig erscheinende wissenschaftliche Publikationen gibt und alle insgesamt betrachtet sehr umfassend sind, wird die Problemstellung anhand einer konkreten Anwendung beispielhaft formuliert. Dabei handelt es sich um das Periodikum ‚inSiDE‘, das im nächsten Unterkapitel vorgestellt wird.

Der Funktionsumfang und die Kompatibilität einer Software unterscheiden sich von Hersteller zu Hersteller. Um mögliche Abläufe besser überprüfen und detaillierter beschreiben zu können, beschränkt sich die in dieser Arbeit betrachtete Software-Landschaft auf das Textverarbeitungsprogramm Microsoft Word (Word), das Layout-Programm Adobe InDesign CS 1 (InDesign) und einen HTML-Editor. Grundsätzlich sind andere geeignete Software-Produkte möglich. Weshalb aber gerade die oben genannten Programme ausgewählt wurden, wird im Folgenden kurz begründet.

Die betrachtete Publikation weist einen ständig wechselnden Autorenkreis auf. Daher sollte die Artikelerstellung so einfach wie möglich gehalten werden. Aufgrund der Popularität von Word ist Publizieren mittels dieser Software inzwischen gängige Praxis und wird auch im akademischen Umfeld weit verbreitet eingesetzt.⁶

Die Software InDesign hat sich seit der Version 2.0 in der Praxis bewährt und wird mittlerweile weit verbreitet als professionelles Layout-Programm eingesetzt. Die Bezeichnung InDesign bezieht sich hierbei auf die Produktversion ‚CS 1‘. Eine neuere Version ‚CS 2‘ ist seit Mai 2005 am Markt. Die zum Zeitpunkt der Erstellung der Arbeit aktuell verwendete und verbreitete Version ist jedoch ‚CS 1‘, auf die sich auch im Folgenden bezogen wird.

Dem Webmaster ist freigestellt mit welchem Editor⁷ er arbeiten möchte, da sich die Handhabung dieser Programme durch das Bearbeiten am HTML-Code sehr ähnelt und im weiteren Verlauf dieser Arbeit für detaillierte Beschreibungen nicht entscheidend ist, welcher Editor ausgewählt wird.

⁶ Vgl. Rothfuss/Ried (2003), S. 31

⁷ Beispiele für HTML-Editoren sind Macromedia Dreamweaver, Adobe GoLive und Microsoft FrontPage etc.

Von dem Anwendungsbeispiel ‚inSiDE‘ und der festgelegten Software-Landschaft ausgehend wird am Ende von Kapitel 2 auf den Anspruch wissenschaftlicher Texte eingegangen und ein entsprechendes Anforderungsprofil erstellt.

2.1 Anwendungsbeispiel: die Fachzeitschrift ‚inSiDE‘

Der Titel der Zeitschrift steht als Akronym für ‚Innovatives Supercomputing in Deutschland‘. Sie wird zweimal jährlich vom Bundeshöchstleistungsrechenzentrum der Universität Stuttgart (HLRS) mit einer Auflage von 2000 Exemplaren publiziert und beinhaltet wissenschaftliche Artikel über aktuelle Projekte und Anwendungen in der Kooperation zwischen den drei Höchstleistungsrechenzentren in Deutschland. Hierbei handelt es sich um das HLRS, das John von Neumann Institute for Computing (NIC) und die Leibniz-Akademie der Wissenschaften (LRZ).

Die Fachzeitschrift repräsentiert zwar alle drei Supercomputing-Zentren, die Produktion und Veröffentlichung erfolgt jedoch ausschließlich am HLRS. Dieses unterstützt Benutzer in Forschung und Anwendung von Supercomputer-Technologien. Sein Anliegen ist es seinen Benutzern mittels der Bereitstellung von Systemen, Werkzeugen und Fachwissen zu einer internationalen Konkurrenzfähigkeit in der Forschung zu verhelfen.⁸

2.2 Ist-Zustand eines beispielhaften Publikations-Workflow

Ein in der Praxis typischer Produktionsablauf erfolgt nach dem Prinzip des traditionellen Publishings:

Konzept – Erstellung/Beschaffung – Aufbereitung/Fertigstellung – Publikation – Distribution.⁹

In Abbildung 2 wird der Ist-Zustand des Beispielproduktionsablaufes, welcher in die Phasen Inhaltsbeschaffung, Print-Veredelung und Web-Veredelung unterteilt ist, veranschaulicht. Dabei verläuft der Print Workflow (hell hinterlegt) nach dem traditionellen Publishing-Prinzip, der Web Workflow (dunkel hinterlegt) beginnt im Anschluss an den Freigabeprozess des Print-Workflow.

⁸ Vgl. HLRS (2006)

⁹ Vgl. Schaubild „Ablaufschema Traditionelles Publishing“, Rothfuss/Ried (2003), S. 25

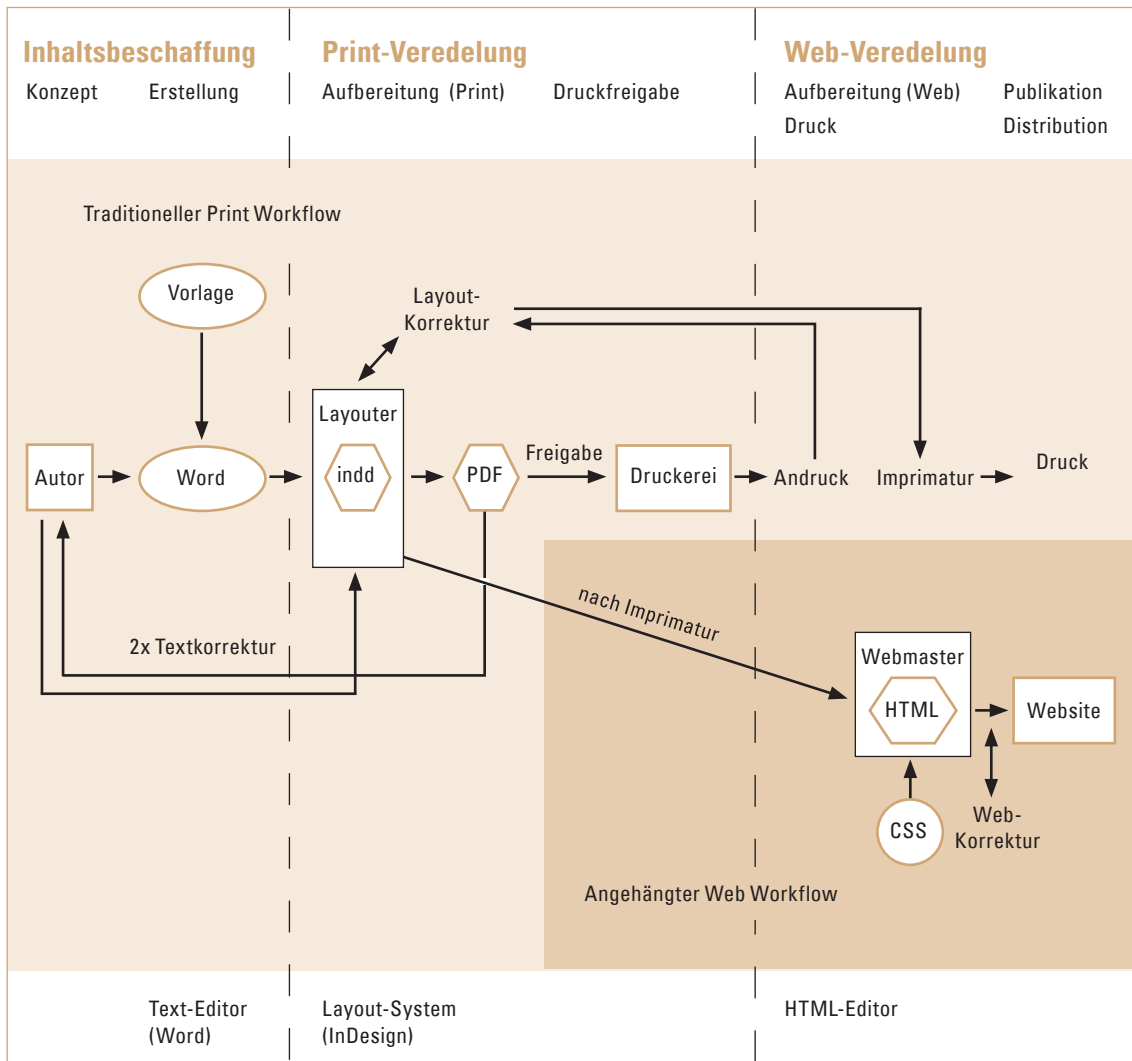


Abb. 2: Schematische Darstellung eines beispielhaften Ist-Zustandes zwei verschiedener Produktionsabläufe

Der Workflow beginnt mit der Veröffentlichung des Abgabetermins für sämtliche Artikel, die in dem Druckprodukt enthalten sein sollen. Die Autoren werden aufgefordert ihre Artikel in Word zu verfassen und dann an die Druckvorstufe weiterzuleiten. Dabei lassen sich beispielsweise mathematische Formeln relativ leicht über einen Formeleditor in Word integrieren und bearbeiten. Abbildungen werden in das Word-Dokument eingebettet oder als separate Bilddateien mitgeschickt – dabei muss im letzteren Fall eindeutig geklärt sein, an welcher Textstelle die Abbildungen platziert werden sollen.

Nach Redaktionsschluss sichtet der Layouter¹⁰ das eingegangene Material und kontrolliert, ob das Bildmaterial für den Druck qualitativ – z.B. die Auflösung oder den Farbraum betreffend – ausreichend ist. Ist dies nicht der Fall, werden qualitativ bessere Bilddaten angefordert. Die

¹⁰ Unter dem Begriff ‚Layouter‘ wird in dieser Arbeit neben dem Schriftsetzer auch der Grafiker verstanden, der gegebenenfalls das Bildmaterial über ein geeignetes Bildbearbeitungsprogramm bearbeiten muss. Der Layouter ist somit für Satz, Gestaltung und Bildbearbeitung verantwortlich.

Artikelreihenfolge wird mit dem Herausgeber abgesprochen. Danach werden die Artikel und Bilder mittels des Layout-Programms montiert und grafisch aufbereitet. Anschließend wird ein PDF-Dokument erzeugt, das den Autoren zur Korrektur vorgelegt wird. Hierzu werden die Texte, das Layout und die Bildplatzierung auf Vollständigkeit, Satz und Fehler kontrolliert. Nach in der Regel zwei Korrekturdurchläufen erfolgt die endgültige Druckfreigabe durch den Verantwortlichen, die Daten gehen an die Druckerei.

Die Druckerei macht einen Andruck zur Imprimatur¹¹. Dies bedeutet die endgültige Druckfreigabe und bietet die letzte Korrekturmöglichkeit. Nach erfolgter Produktion kommt das fertige Druckprodukt zur Distribution an den Auftraggeber zurück oder geht an einen Distributor.

Bei einer parallelen Veröffentlichung des Mediums im Internet erhält der Webmaster, der für den kompletten Workflow der Publikation im Internet verantwortlich ist, die freigegebene Druckvorlage der Zeitschrift, um daraus die benötigten Inhalte zu extrahieren. Sobald die Imprimatur erfolgt ist, beginnt die entsprechende Umsetzung innerhalb des festgelegten Zeitraumes. Falls das Produkt zum Zeitpunkt der Distribution auch online publiziert werden soll, müssen die kompletten Daten bis zur Fertigstellung des Druckprodukts für das Web aufbereitet sein.

Der Web Workflow verläuft folgendermaßen: Das druckfertige PDF und die offene Satzdatei sowie das Bildmaterial gehen an den Webmaster. Dabei ist es ihm überlassen, ob er die neuen Inhalte anhand des PDF-Dokuments oder der Satzdatei in die vorhandene Website integriert. Die Website ist statisch mit Vorlagen aufgebaut. Der Webmaster bereitet die Online-Ausgabe mittels einer neuen Vorlage vor, per Copy&Paste- oder Import-Funktion werden die Texte in die HTML-Dokumente, die mit der passenden Vorlage erstellt wurden, übertragen. Hierbei können satztechnische Schwierigkeiten auftreten, die in dem nachfolgenden Unterkapitel näher erläutert werden. Des Weiteren müssen die Verlinkungen angepasst werden, wenn beispielsweise die vorherige Ausgabe in das Archiv gestellt wird, weil nunmehr die aktuelle Ausgabe erscheint. Arbeitet man hingegen mit einem HTML-Editor und entsprechenden Vorlagen, werden die Verlinkungen größtenteils vom Editor automatisch aktualisiert. Dennoch finden sich immer wieder Bereiche, die manuell aktualisiert werden müssen.¹²

Die Abbildungen werden separat aus den Referenzen der Satzdatei herausgesucht, über einen Bildeditor Web-gerecht¹³ bearbeitet, gespeichert und der relevanten Stelle des jeweiligen Artikels hinzugefügt. Aufgrund des begrenzten Platzes auf dem Bildschirm werden die Schaubilder

¹¹ Die Imprimatur (lat. es werde gedruckt) ist die Druckerlaubnis nach dem Korrekturlesen, die durch den Autor, Verleger oder Herausgeber erteilt wird

¹² Vorlagen sind HTML-Seiten oder Skripte, die ein Gerüst aus Platzhaltern bilden und die auf HTML-Dateien angewandt werden. Werden Änderungen an dem Gerüst vorgenommen, so werden sie in den Vorlagen-basierten HTML-Dateien automatisch übernommen, die individuell gestalteten Bereiche müssen in jeder HTML-Datei manuell geändert werden. Vgl. Büchner/Traub/Zahradke/Zschau (2001), S. 198

¹³ Ein Web-gerechtes Bild hat eine Auflösung von 72-92 dpi und ist somit für eine Bildschirmdarstellung optimiert.

und Grafiken oft relativ klein abgebildet, so dass dem Leser aus Gründen der Lesbarkeit eine größere Version zur Verfügung gestellt werden muss. Dies geschieht, indem erstens ein kleineres Web-gerechtes Bild, das eine durch das Website-Layout begrenzte Größe nicht überschreiten darf, in den Text eingebettet wird und zweitens ein größeres Bild erstellt wird, das auch eine Detailwiedergabe des Bildes gewährleistet. Das größere Bild kann in einem neuen Fenster geöffnet werden, für das eine neue HTML-Datei erstellt wird. Ist die Web-Aktualisierung abgeschlossen, wird sie auf dem Webserver zur Online-Publikation bereitgestellt. Korrekturen finden nur über den Webmaster anhand der Druckversion statt.

2.3 Zustandsanalyse

Der vorliegende Workflow gleicht dem optimalen¹⁴ Publikationsablauf eines gedruckten Mediums, das anschließend zusätzlich im Internet publiziert werden soll. Da bisweilen die gedruckte Ausgabe bereits vor der Online-Publikation existiert und somit die Entscheidung für die Online-Publikation erst nachträglich gefallen ist, ist dieser Ablauf keineswegs optimal angelegt. Der aufbereitete Datenbestand ist nicht darauf ausgerichtet, sich ohne weiteres für andere Zwecke anpassen zu lassen. Verwertbare Vorarbeiten, wie aufbereitete Daten, sowie Zwischenergebnisse, die sich ohne einen Zusatzaufwand anderweitig nutzen lassen, sind derzeit nicht vorhanden. Die Erstellung und Bearbeitung einzelner Objekte lässt sich für Außenstehende nicht nachvollziehen. Nur die an der Produktion beteiligten Personen wissen, welche Inhalte wie, wo, wann und warum in die Publikation eingeflossen sind und wie sie bearbeitet wurden. Es handelt sich also um eine Produktion ohne nennenswerte Automatisierung.^{15,16}

Auftretende Probleme und Schwachstellen dienen als Grundlage für die Ableitung von Verbesserungsmaßnahmen. Welche Nachteile der Status quo aufweist, die insbesondere den Web Workflow ineffizient und mühsam machen, wird im Folgenden dargelegt:

1. Fehlerquellen

Fehler können dort entstehen, wo die aktive Bearbeitung von Daten durch Mitarbeiter gefordert wird, d.h. beim Layouter oder bei der Übertragung der Daten in das Web Layout. Dabei können vor allem nach der Druckfreigabe durch den Webmaster bedingt erneut Fehler verursacht werden, die vorher bereits korrigiert wurden oder gar nicht existierten. Dies passiert beispielsweise

¹⁴ Der Print Workflow ist insofern optimal, als dass er zu einem sehr zufrieden stellenden Endprodukt führt und lediglich bei der Übertragung der Inhalte – besonders der mathematischen Ausdrücke – in das Layout-Programm Verbesserung erlangen kann.

¹⁵ Ob es wünschenswert ist, den gesamten Workflow in dieser Hinsicht zu automatisieren, wie es in der Mehrfachverwendung durch strukturierte Inhalte in modernen Publishing-Auffassungen der Fall ist, bleibt fraglich. Im Web Workflow ist ein einheitliches Layout durchaus von Vorteil. Es erleichtert die Orientierung auf der Website oder einem alternativen elektronischen Medium. Im Print Workflow überwiegen zwar ebenfalls wiederkehrende Abläufe – beispielsweise bei einer periodisch erscheinenden Zeitschrift – jedoch hat die Praxis gezeigt, dass hier im Gegensatz zu Veröffentlichungen im Internet eine gewisse gestalterische Freiheit durchaus erwünscht ist.

¹⁶ Vgl. dazu Rothfuss/Ried (2003), S. 24ff

durch das Einkopieren der Texte in die HTML-Dateien, wobei versehentlich Zeichen gelöscht werden können oder nicht mit ausgewählt und übertragen werden.

2. Formeldarstellung

Der Formelsatz gestaltet sich schwieriger als das Setzen eines Textes, da die Zeichen der Formeln nicht nur nebeneinander, sondern auch übereinander stehen (z. B. bei Brüchen). Manche Zeichen müssen sich über nachfolgende Zeichen erstrecken (z. B. Wurzeln) oder ihre Größe abhängig von nachfolgenden Zeichen verändern (z. B. Klammern). Es besteht das allgemeine Problem, Formeln korrekt darstellen und übergeben zu können. Zunächst sollte die Formeldarstellung im Print Workflow betrachtet werden, danach analog im Web Workflow: In Word ist es über den Formel-editor möglich, Formeln einzugeben und sie auf einfache Weise zu editieren. Importiert man den gesamten Text mit Formeln jedoch in das verwendete Layout-Programm, wird eine Formel als ‚Bildobjekt‘ definiert und nicht mehr als ‚Formel‘ behandelt. Die einzelnen Zeichen lassen sich nicht mehr verändern und somit kann auch z.B. die Schriftart oder der Zeichenabstand nicht mehr korrigiert werden. Eine weitere Methode, Formeln in Word hinzuzufügen, stellt die ‚Symbol Einfügen‘-Funktion dar. Die einzelnen mathematischen Sonderzeichen werden in der Schriftart ‚Symbol‘ dargestellt, die beim Import in das Layout-Programm korrekt übernommen werden. Werden die Texte allerdings mittels Copy&Paste-Funktion dem Layout-Programm hinzugefügt, werden auch die Sonderzeichen der Schriftart ‚Symbol‘ nicht mehr korrekt übernommen, und es werden lediglich Platzhalter eingefügt. Bei dieser Methode ist generell problematisch, dass nicht alle für mathematische Ausdrücke notwendigen Zeichen in dieser Schriftart enthalten sind und somit einige Formeln auf diese Weise nicht dargestellt werden können und keine korrekte Darstellungsform möglich ist. Eine Nacharbeitung der Formeln im Layout-Programm wird notwendig, über die eine korrekte Darstellungsform durch den Einsatz verankerter Rahmen und entsprechenden erweiterten Schriftsätzen oder einer Bildgenerierung über ein entsprechendes Hilfsprogramm gewährleistet werden kann.¹⁷

Bei der Übertragung der Formeln in die HTML-Dateien mittels der Funktion Copy&Paste werden die Zeichen in der Schriftart ‚Symbol‘ ebenso nur bedingt korrekt übertragen, d.h. es werden lediglich die in der HTML-Spezifikation vordefinierten Sonderzeichen abgebildet. Die Zeichen, die in der ASCII-Zeichentabelle nicht hinterlegt sind, werden durch Platzhalter ersetzt. Sie können über einen Bildeditor zu kleinen GIF-Dateien umgewandelt und an der gewünschten Textstelle eingefügt oder – seit der Unicode-Standardisierung¹⁸ – manuell über die Unicode-Zeichentabelle korrigiert werden. Formeln, die in Word über den Formeleditor erstellt wurden, werden im HTML-Editor als GIF-Bilddatei eingefügt. Dadurch entstehen Darstellungs- und Kodierungsprobleme;

¹⁷ Vgl. Forssman/de Jong (2002), S. 202 f

¹⁸ Das Unicode-System beschreibt die Zeichen und Elemente aller bekannten Schriftkulturen und Zeichensysteme. Durch den international standardisierten Zeichensatz reicht es aus, von der ISO 8859-1 auf UTF-8 umzuschalten, damit die Sonderzeichen im Unicode dargestellt werden können. Viele Seiten beruhen allerdings noch immer auf der ISO-Norm.

so muss beispielsweise bei der Bearbeitung einer Formel diese jedes Mal vollständig neu generiert werden. Darüber hinaus kann die Formel nicht exakt mit dem Text auf eine Grundlinie gebracht werden, weil die verschiedenen bei den Nutzern des Internets in Gebrauch befindlichen Browser-Programme Unterschiede in der Darstellungswiedergabe aufweisen. Deshalb werden die oben genannten Elemente auch nicht als Formeln, sondern als Bilder gehandhabt, wodurch keine Verwertbarkeit der Formel in anderen Programmen möglich ist.

Ein weiteres Problem stellen die hoch- und tiefgestellten Zeichen dar, die bei der Übertragung nicht erkannt werden. Die Texte müssen also Schritt für Schritt mit der Druckausgabe verglichen und auf mögliche hoch- bzw. tiefgestellte Zeichen und Formeln mit Sonderzeichen kontrolliert und ggf. manuell bearbeitet werden.

3. Bilddaten

Bilder helfen dem Leser Textinhalte zu veranschaulichen und sind fester Bestandteil von Publikationen. Betrachtet werden soll hier nicht die Erstellung der Bilddaten, sondern die Frage, wie sie in ein Text-Dokument integriert und anschließend an die DTP-Vorstufe übergeben werden. Die Übergabe der Bilddaten vom Autor zum Layouter (DTP-Vorstufe) ist insofern kritisch, als viele Autoren ihre Bilder in das Text-Dokument einbetten. Fügt man sie über die Funktion ‚Grafik-Einfügen‘ ein, so werden die Bilder in den Word-Code eingebettet. Bei anschließender Bearbeitung des Bildes wird der eingebettete Code unwiderruflich verändert und das Ursprungsformat des Bildes ist somit nicht mehr gegeben.¹⁹ Dadurch kann die Bildqualität gemindert und somit für den Druck unbrauchbar werden.

Bilddaten können auch separat als Bilddatei übermittelt werden. Der Nachteil hierbei ist jedoch, dass vom Autor nicht immer Informationen über die gewünschte Position der Bilder im Text existieren. Zudem ist naturgemäß die Gefahr größer, dass der Verfasser des Beitrages die Über-sendung der Bilddateien vergisst. Ein weiteres Problem, das höchstwahrscheinlich mit der Zeit auftreten wird, stellt die Bilddatenverwaltung dar. Die Dateien werden sich von Ausgabe zu Ausgabe vervielfachen und Speicherplatzbedarf verursachen, daher sollte die Redundanz der Daten möglichst gering gehalten werden.

4. Satz

Satz- bzw. Formatierungsprobleme gibt es vor allem in der Textübertragung von der druckfertigen Variante zur Web-Variante. Beispielsweise sind so genannte ‚weiche‘ Absätze vorhanden,²⁰ die in HTML nicht erkannt werden, oder es werden Absätze hinzugefügt, die in der Web-Darstellung nicht vorhanden sein sollen. Verborgene Satzzeichen stellen folglich eine besonders kritische Fehlerquelle für die Übertragung von Print-Darstellungen auf Web-Darstellungen dar.²¹

¹⁹ Dies kann z.B. dann passieren, wenn ein Autor die Bildgröße verändert, um sie an den Seitentext anzupassen.

²⁰ Ein „weicher“ Absatz führt nur zu einem Zeilen-Umbruch, nicht zu einem vollständigen neuen Absatz.

²¹ Zu den verborgenen Satzzeichen zählen Absatzmarken, Tabulatoren, etc.

Ein wesentlicher Grund hierfür ist, dass in der Druckvariante der Text einmalig und statisch auf die Seitengröße angepasst werden muss. In der Web-Darstellung wird das automatisch durch den Browser vorgenommen, d.h. wenn der Schriftgrad in der Browser-Ansicht beispielsweise auf kleine Schrift eingestellt ist, wird ein Zeilenumbruch im Text später notwendig als bei größerer Schrifteinstellung. Auch die Überlegung, den Text der Druckvariante zunächst formatierungsfrei zu übertragen, hilft nicht weiter; denn fehlen die notwendigen Absätze, welche den Text für den Leser übersichtlich machen sowie inhaltlich voneinander abgrenzen, wäre auch hier ein erneuter manueller Satzvergleich die notwendige Folge.

2.4 Ziel und Aufgabenstellung

Die Qualität eines Workflow richtet sich nach den Kriterien Prozesssicherheit, Wirtschaftlichkeit und Qualität. Prozesssicherheit bedeutet, den Ablaufprozess zu vereinfachen und sicher zu gestalten. D.h. er muss genau definiert und spezifiziert sein und kann gegebenenfalls durch einen Automatismus unterstützt werden. Wirtschaftlichkeit wird über eine Minimierung der entstehenden Kosten sowie eine Maximierung der Verwertbarkeit erzielt, die durch strukturierte Inhalte gegeben ist.²² Qualität wird einerseits über eine optimale Prozesskontrolle, andererseits über eine gute Produktveredelung erreicht.

Um die oben genannten Ziele erreichen zu können, müssen bestimmte Bedingungen erfüllt sein: Inhalte müssen strukturiert werden, um einen Mehrwert der Daten zu erhalten; es muss ein definierter Vorgang vorliegen, der für die Transformation der Inhalte in die Ausgabemedien zuständig ist; und vor allem sind die Ziele unter dem Aspekt der Umsetzung von wissenschaftlichen Texten zu realisieren, wodurch nicht nur die Frage nach Integration der medienneutralen Datenhaltung und strukturierten Datenaufbereitung im vorhandenen Workflow-Prozess gestellt ist, sondern auch wie diese unter dem Gesichtspunkt der Verarbeitung von Sonderzeichen und mathematischen Formeln am besten erreicht wird. Dabei ist zu berücksichtigen, dass die Inhalte nicht nur aus Texten bestehen, sondern auch Bilder und Grafiken enthalten, die gesondert behandelt werden müssen. Des Weiteren sollen unter dem Gesichtspunkt der Prozesskontrolle die Korrekturschritte minimiert und so integriert werden, dass die Weiterverarbeitung der strukturierten Daten keine inhaltlichen Fehlerquellen mehr zulässt.

²² Strukturierte Inhalte sind Daten, deren Inhalte von ihrer Gestaltung getrennt vorliegen. Durch die Struktur lassen sich Dokumentenbestandteile eindeutig beschreiben. Bei geeigneter Anwendung kann darüber ein Dokument automatisiert transformiert werden. Beispielsweise kann durch einen generischen Software-Agenten das Dokument erschließbar gemacht werden (Prinzip der Suchfunktion). Vgl. dazu Rothfuss/Ried (2003), S. 61

2.5 Anforderungsprofil

2.5.1 Anspruch wissenschaftlicher Texte

Wie bereits aus den Schwächen des Status quo ersichtlich stellt die Darstellung von mathematischen Formeln einen besonderen Anspruch an eine Publikation dar; und zwar sowohl an das Print- als auch an das Web-Produkt, jedoch in unterschiedlicher Form des Formelsatzes, wie im Folgenden erläutert wird.

Mathematische Funktionen und Ausdrücke sowie technische Symbole manifestieren häufig den inhaltlichen Kern des Beitrags. Deren Darstellung fußt jedoch nicht auf den Zeichen des ASCII-Zeichensatzes, sondern auf jenen Sonderzeichen, die in HTML nur über die Unicode-Adressierung und in der Druckvorstufe durch entsprechende Schriftsitzerweiterungen wiedergegeben werden können.²³ Unter einem Zeichensatz wird dabei der Vorrat an Elementen zur Darstellung von Sachverhalten verstanden (z.B. Unicode oder ASCII). Als Schriftart wird in der Typografie die grafische Gestaltung eines Zeichensatzes bezeichnet.

Der Unicode-Standard muss für eine Wiedergabe in der HTML-Datei explizit über eine Referenz definiert sein. Standardmäßig wird in vielen HTML-Editoren noch die ISO 8859-1 Zeichentabelle²⁴ verwendet, die zwar viele Sonderzeichen der westeuropäischen Sprache beinhaltet, allerdings nicht alle, die in mathematischen Funktionen ihre Anwendung finden.

Neben der Unicode-Zeichenkodierung gibt es im Internet weitere Darstellungsmöglichkeiten von Formeln. Einen Text, der Formelzeichen enthält, als Grafikdatei zu speichern und komplett als Grafikseite abzubilden, stellt jedoch keine gute Lösung dar, weil die Datei durch ihre Größe unhandlich für den Zugang im Web wäre. Häufig – wie auch im Status quo ersichtlich – werden Formelzeichen als kleine Bilddateien in HTML eingebettet oder im nativen HTML mit der Schriftart ‚Symbol‘ dargestellt. Bedenklich dabei ist, dass nach einer Bearbeitung der Formel das Bild komplett neu generiert werden muss.

Seit 2001 wurde vom W3C-Gremium die Auszeichnungssprache MathML²⁵, die zur XML-Familie zählt, standardisiert. Sie soll mathematische Gleichungen so beschreiben, dass Software-Programme diese verstehen und anwenden können. Dafür werden die Inhalte in ihrer logischen

²³ Nicht alle Schriftsätze enthalten alle Sonderzeichen. Schriftsätze, die mathematische Formelzeichen und weitere Sonderzeichen enthalten, basieren teilweise auch auf dem Unicode-Standard (OpenType Schriften); andere wurden speziell im Schriftdesign um entsprechende Sonderzeichen erweitert.

²⁴ ISO 8859-1, genauer ISO/IEC 8859-1, ist ein von der ISO zuletzt 1998 aktualisierter Standard für die Informationstechnik zur Zeichenkodierung mit acht Bit. ISO 8859-1 versucht möglichst viele Sonderzeichen westeuropäischer Sprachen abzudecken. Dennoch sind z.B. Währungszeichen und viele andere Sonderzeichen, die in mathematischen Formeln Anwendung finden, nicht in diesem Standard vorhanden. Dazu wurden weitere ISO-Normen geschaffen, die jedoch explizit im Browser einzustellen bzw. in HTML-Dateien einzubinden sind.

²⁵ Vgl. W3C (2006)

Struktur unabhängig von ihrer grafischen Gestaltung abgelegt. MathML-kodierte Ausdrücke müssen jedoch interpretiert werden, um eine grafische Abbildung in einem Anwendungsprogramm darstellen zu können; dies kann über ein geeignetes PlugIn ermöglicht werden. Zur Darstellung über einen Webbrowser werden solche PlugIns in Form eines Reader oder eines Player zum kostenlosen Download zur Verfügung gestellt.²⁶

2.5.2 Anforderungen

Es werden sieben separate Anforderungen identifiziert und nachfolgend näher charakterisiert. Die hier vorgestellten Anforderungen eins bis drei, die sich aus dem bestehenden Prozess ergeben, beziehen sich auf die Software-Landschaft sowie den Gestaltungsanspruch, der für die Produktqualität ausschlaggebend ist. Die Anforderungen vier bis sieben bezeichnen die Merkmale, die einen neuen Workflow-Prozess gestalten und für das Erreichen der Ziele notwendig sind. Sie werden aus den Problemen und Schwierigkeiten im Status quo hergeleitet und beeinflussen die Datenbeschaffung, den Datenaustausch und die Datenveredelung. Die Anforderungen eins bis sieben lassen sich folgendermaßen im Detail beschreiben.

1. Software-Landschaft

Die Software-Landschaft bezieht sich auf die Arbeitswerkzeuge der am Workflow beteiligten Personen; damit sind die Anwendungsprogramme der Redaktion, Druck- und Web-Vorstufe gemeint. Wie im Status quo beschrieben, wird den Autoren nahe gelegt, ihre Artikel in Form von Word-Dokumenten an die DTP-Vorstufe zu übergeben. Das impliziert, dass Workflows, die von einem anderen Programm zur Inhaltserstellung ausgehen, in dieser Arbeit im Weiteren nicht berücksichtigt werden. Dies stellt damit eine zentrale Anforderung an die Gestaltung der Lösungswege dar. Analog dazu gilt die Wahl der Arbeitsumgebung des Layouters. Eine Einschränkung der Werkzeuge der Web-Vorstufe wird nicht vorgenommen, da diese durch den möglichen Einsatz eines Automatismus für diese Arbeit an Bedeutung verlieren.

2. Gestaltung

Eine Publikation hat einen ganz besonderen Anspruch an die Gestaltung, der über das folgende Zitat sehr gut verdeutlicht wird:

„Bei allen Publikationen, die für einen menschlichen Benutzer gedacht sind, ist die Präsentation der Inhalte, also ihre professionelle Gestaltung, wesentlicher Bestandteil des Herstellungsprozesses und verbraucht einen erheblichen Teil seiner Ressourcen [...] Die professionelle Gestaltung von Inhalten gehört zu den anspruchsvollsten Tätigkeiten des Publishing überhaupt, da sie typographisches, graphisches, zeichnerisches und photographisches Kunsthandwerk mit Wirtschaftlichkeit verbinden muss. Sie ist von zwei großen Widersprüchen gekennzeichnet: dem

²⁶ Vgl. W3C (2006)

Widerspruch zwischen Perfektion und Rationalisierung bzw. Automation einerseits und dem Widerspruch zwischen Struktur und Präsentation andererseits.“²⁷

Prinzipiell können zwar durch ein automatisiertes Layout auf Knopfdruck ganze Kataloge erstellt werden. Aber, wie das Zitat schon andeutet, sind in der Regel Automation und Präsentation widersprüchliche Bedingungen. Ein voll automatisierter Herstellungsprozess lässt eine individuelle Gestaltung des Layout nicht zu und bringt daher immer das gleiche Aussehen hervor.²⁸ Bei einer Katalogproduktion mag das durchaus gewollt sein. Eine Fachzeitschrift, dessen Erscheinungsbild unterschiedlichen gestalterischen Ideen des Layouters unterworfen ist, variiert optisch von einer Ausgabe zur Nächsten – dies gilt, wie oben erläutert wurde, vor allem für ein gedrucktes Medium. Das Web-Produkt verfolgt andere Ziele, die wichtiger sind als eine ansprechende Gestaltung. Eine Website, die eine hierarchisch vernetzte Struktur aufweist, muss vor allem funktional und übersichtlich für den Benutzer sein, damit er schnell zu der benötigten Information gelangt. Ein Druckprodukt wird für gewöhnlich linear gelesen und benötigt keine Hilfe von Suchfunktionen und verlinkten Verweisen, um auf seine Inhalte zugreifen zu können. Durch eine ausdrucksvolle Gestaltung kann sich zudem eine Fachzeitschrift von anderen abheben, was gerade für den Herausgeber von besonderem Interesse ist, da eine vorteilhafte Gestaltung z.B. gegenüber – potentiellen – Werbekunden als Wettbewerbsvorteil benutzt werden kann.²⁹ Ein automatisierter Print Workflow ist mithin nur dann erwünscht, wenn es gelingt, in dieser Automatisierung eine Möglichkeit zu schaffen, partiell in das Layout eingreifen zu können. Aus technischer Sicht zeigt sich, dass ein sich laufend veränderndes Erscheinungsbild schwer über Regeln einer Formatierungssprache zu definieren ist.³⁰ Für jede Ausgabe müssten die Formatierungsanweisungen angepasst werden, wobei durch den manuellen Einfluss keine Automatisierung mehr gewährleistet werden würde. Es ist möglich und von Vorteil, das Web-Produkt automatisiert zu erstellen; das Druckprodukt sollte jedoch dem traditionellen Bearbeitungsweg über ein Layout-Programm folgen.

3. Qualitätssicherung

Die Qualität des Web-Produktes wird durch einen automatisierten Web Workflow sichergestellt. Dadurch wird die Anzahl möglicher Fehlerquellen minimiert.

²⁷ Zitat aus Rothfuss/Ried (2003), S. 29 f

²⁸ „Mit automatischen Mengensatzmethoden lassen sich keine ästhetisch perfekten Satzergebnisse erzielen. Das liegt an der Unmöglichkeit, alle praktisch vorkommenden Kombinationen von Text und Bild, Textmengen in verschiedenen Boxen und Spalten, Größenunterschiede von Bildern in einer Bildstrecke und dergleichen in den Regeln eines automatischen Satzprogramms abzubilden, zumal sich die verwendeten ästhetischen Grundsätze von Layout zu Layout unterscheiden. Außerdem lebt ein gutes Layout von subtilen Feinheiten der Platzierung und Balance, die ein automatischer Prozess schon deswegen nicht erzeugen kann, weil er gleichartige Objekte gleichartig behandelt.“ Zitat aus Rothfuss/Ried (2003), S. 30

²⁹ „[...] der Wettbewerb im Markt ist so sehr stark und die Ansprüche der Zielgruppen sind so hoch, dass optimale Gestaltung als wichtiger Wettbewerbsvorteil genutzt werden muss.“ Zitat aus Rothfuss/Ried (2003), S. 30

³⁰ Aus einem erzeugten XML-basierten Datenbestand kann mittels der Formatierungssprache XSL-FO automatisch ein drucktaugliches PDF generiert werden.

Um jedoch die Qualität des Print-Produktes zu sichern, muss der Freigabezyklus, also die Korrekturschritte Text-, Layout-, Farb- und Druckbogenkontrolle, weiterhin eingehalten werden. Auch eine sinnvolle Verteilung der Korrekturschritte kann eine Vermeidung bzw. Minimierung von Fehlerquellen zur Folge haben. Beispielsweise muss es nach der Artikelfreigabe keine Möglichkeit mehr geben, diesen wieder verändern zu können. Darüber hinaus zeichnet sich die Qualität des Druckmediums insbesondere dadurch aus, dass der Gestaltungsanspruch garantiert werden muss. Hebt sich das Layout gestalterisch nicht mehr ab, wird die Qualität gemindert.

4. Formelerstellung und Formelverarbeitung

Die Erstellung und Verarbeitung von Formeln soll in dieser Arbeit gesondert betrachtet werden. Dem Autor soll eine einfache und intuitiv gestaltete – d.h. eine weitgehend selbsterklärende – Eingabemöglichkeit für mathematische Ausdrücke empfohlen werden, die gleichzeitig der DTP-Vorstufe, die diese Daten verarbeiten muss, eine geeignete Möglichkeit bietet die Formelausgaben sinnvoll zu verwerten. Eine fehlerlose Formelübertragung von einer Produktionsumgebung zur nächsten muss also gewährleistet sein und hängt von der Kodierungsart ab. Die Notwendigkeit, die Darstellung der Sonderzeichen auf Korrektheit zu überprüfen, sollte entfallen. Platzhalter, die durch nicht korrekte Definition der Zeichen im Status quo vermehrt aufgetreten sind, dürfen nicht mehr vorkommen.

5. Dokumentübergabe von Autor zur DTP-Vorstufe (Datenaustausch)

Ein erstellter Artikel muss in Form eines Dokumentes von dem einzelnen Autor zu der DTP-Vorstufe gelangen. Ein geeignetes Austauschformat dafür basiert auf dem XML-Standard³¹. Dabei müssen Inhalt und Darstellung voneinander getrennt werden. Der Inhalt wird über die XML-Auszeichnung strukturiert, die Darstellung wird über Gestaltungsanweisungen in einem Stylesheet definiert.

Satzprobleme – wie sie im Status quo auftreten – werden dadurch bei der Übergabe zur nächsten Produktionsumgebung vermieden. Es ist jedoch darauf zu achten, dass eine echte Trennung von Inhalt und Darstellung vorhanden ist. In Word beispielsweise ist die Formatierung des Textes zwar eine Strukturierungsmöglichkeit, sie wird jedoch nicht getrennt verwaltet, sondern ist im Word-Dokument eingebettet.³² Eine echte Trennung wird durch den Einsatz von XML ermöglicht. Der XML-Standard findet grundsätzlich zwar nur bei Textinhalten Anwendung; inwieweit aber mit XML die Integration von Formeln und Bildern möglich ist, wird in den Lösungswegen eingehend behandelt.

6. Datenverwaltung

Die verschiedenen Inhalte sollen entsprechend ihrer Kategorie effizient verwaltet werden können, um eine problemlose Auffindung und Weiterverarbeitung zu ermöglichen. Ein Effizienzkriterium

³¹ „Die anfangs vor allem in umfangreichen Dokumenten benutzte Sprache dient mittlerweile im kommerziellen Umfeld weithin als Datenaustauschformat.“ Zitat aus Heise online (2006)

³² Vgl. Rothfuss/Ried (2003), S. 65 f

bildet sich aus der Vermeidung von Content-Mehrfacherfassung, die durch eine medienneutrale Datenhaltung erreicht wird. Weitere Kriterien sind die Minimierung von Speicherplatz, die Reduzierung von Redundanzen, die Einführung einer Zugriffsregelung und Datensicherheit sowie der Nachvollziehbarkeit der Aktualität der Inhalte. Diese Merkmale werden durch den Einsatz eines effizienten Datenverwaltungssystems berücksichtigt.

7. Kostenminimierung

Die Anschaffungskosten für Soft- und Hardware sollen minimal sein. Zwar sind zusätzliche Kosten durch den veränderten Workflow nicht ausgeschlossen, aber natürlich sollten diese so gering wie möglich gehalten werden. Ebenso sollen die Personalkosten möglichst nicht steigen. Der bisher beteiligte Personenkreis soll nicht durch Spezialisten erweitert werden müssen; im Idealfall sind nur Autoren, ein Layouter und ein Webmaster am Workflow beteiligt.

3 Evaluation Teil I – Vorstellung der Evaluationsobjekte

„Evaluieren heißt, bestimmte Objekte auf der Grundlage eines Systems von Evaluierungskriterien zielbezogen zu beurteilen. [...] Ein Evaluierungsbedarf besteht besonders dann, wenn es sich bei den Aufgaben nicht um Routineaufgaben, sondern um Vorhaben handelt, die aus dem üblichen betrieblichen Geschehen herausragen...“³³ Diese Definition bezogen auf die hier behandelte Thematik betrifft die Publikationserstellung aus einem medienneutralen Datenbestand, der heutzutage zwar schon beinahe zur Routinetätigkeit gehört; die cross-mediale Umsetzung wissenschaftlicher Texte durch die Darstellung und Verarbeitung mathematischer und technischer Symbole stellt jedoch einen Sonderfall dar, der spezifisch zu behandeln ist.

Die Evaluationsschwerpunkte werden in dieser Arbeit auf das Lösen der Probleme im Status quo gelegt, um eine effektive Integration eines automatisierten Web Workflow zu schaffen sowie der Anforderung an die Verarbeitung wissenschaftlicher Texte zu genügen.

In den folgenden Unterkapiteln werden verschiedene Vorgehensweisen aufgezeigt. Die Darstellung möglicher Lösungen erfolgt, indem von der Inhaltsbeschaffung über den Datenaustausch bis hin zur Produktveredelung alle Szenarien erörtert werden. Hierbei wird modular auf die Prozessphasen eingegangen. Die einzelnen Module bestehen aus mehreren Elementen, die eine speziell genannte Vorgehensweise beschreiben. Die Auswahl jeweils eines Elementes aus den verschiedenen Modulen setzt sich dann zu einem kompletten Lösungsweg zusammen.

³³ Zitat aus Heinrich/ Lehner (2005), S. 393

In Kapitel 3.1 werden die verschiedenen Module der Inhaltsbeschaffung ‚Formelerstellung‘ und ‚Bildintegration‘, in Abbildung 3 rot eingefärbt, vorgestellt. Die jeweiligen Elemente sind in Abbildung 3 beige eingefärbt. Prinzipiell wird von zwei Erstellungsmöglichkeiten ausgegangen; über Word und über eine Web-Maske. Die genannten Eingabemöglichkeiten von Formeln werden dabei detailliert behandelt und gehen speziell auf den Anspruch wissenschaftlicher Texte ein.

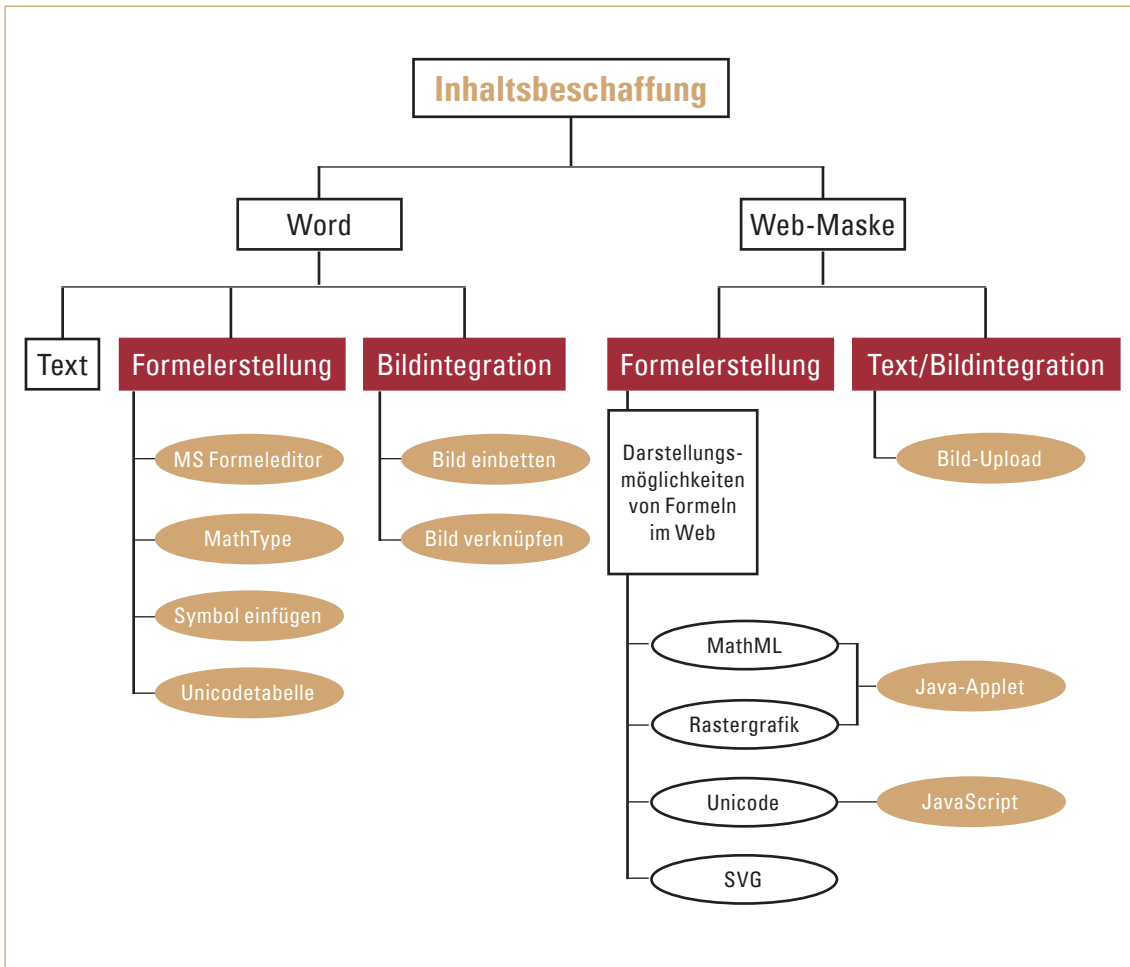


Abb. 3: Schematische Darstellung der Inhaltsbeschaffung

In Kapitel 3.2 werden die verschiedenen Möglichkeiten des Datenaustausches in Form der Datenübergabe von Autor zur DTP-Vorstufe festgelegt. Der Austausch soll auf Basis der XML-Spezifikation erfolgen. Die einzelnen Elemente, in Abbildung 4 rot dargestellt, zeigen die Möglichkeit der XML-Erzeugung auf. Auf die Integration von Formeln und Bilder in die XML-Auszeichnung wird in Kapitel 3.3. eingegangen.

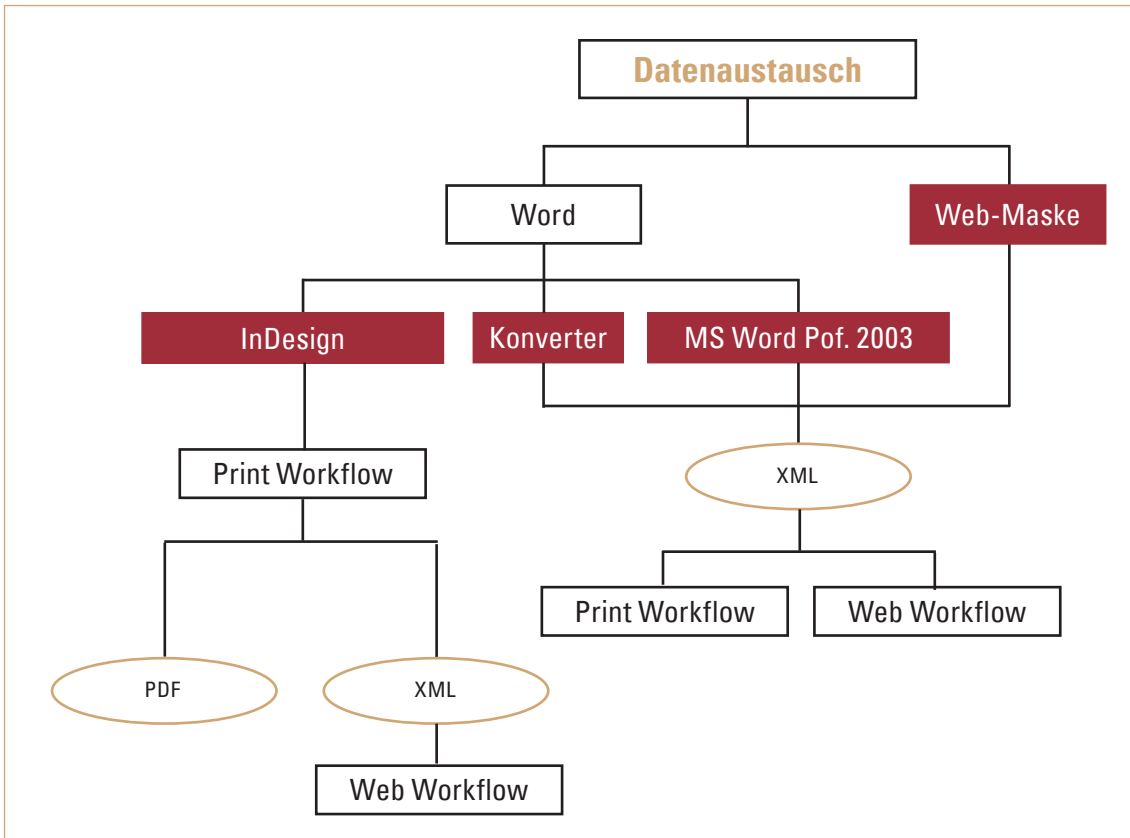


Abb. 4: Schematische Darstellung des Datenaustausch und des Prozessablaufes von Print und Web

Die unterschiedlichen Ablaufreihenfolgen eines Workflow – parallele versus sequentielle Produktveredelung – werden im Anschluss daran in Kapitel 3.4 erörtert. Die Datenaufbereitung lässt sich grundsätzlich in zwei Arten von Prozessabläufen aufteilen: Zum einen ist eine Integration des Web Workflow in den gesamten Produktionsablauf durch die Schaffung eines medienneutralen Datenbestandes möglich. Dies hätte parallele Fertigungsprozesse von Print- und Web-Produkt zur Folge (s. Abbildung 5, oben). Zum anderen ist die Fertigstellung des Print-Produkts mit anschließender Integration des Web Workflow denkbar, wodurch sich ein sequentieller Ablaufprozess ergeben würde (s. Abbildung 5, unten).

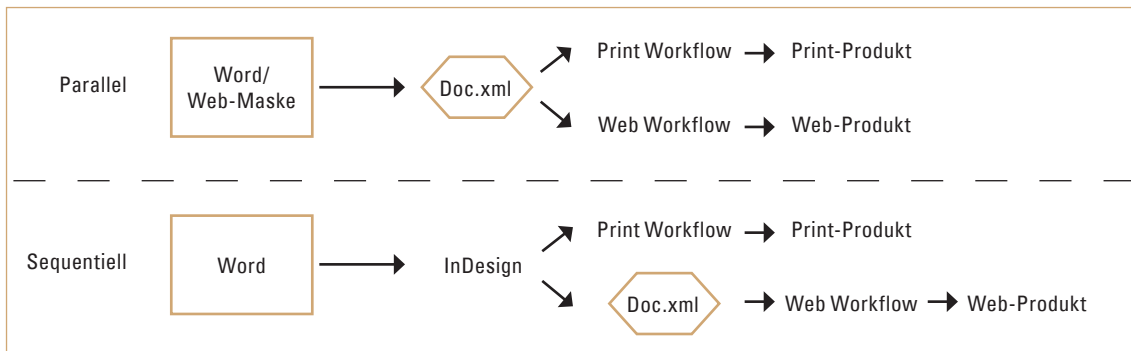


Abb. 5: Parallel versus sequentiell ablaufender Workflow-Prozess

Auf einen Lösungsweg in Form einer System-Lösung wird in Kapitel 3.5 eingegangen und begründet, warum dieser nicht weiter verfolgt wird. Am Ende des ersten Teils der Evaluation wird ein Kriterienkatalog erstellt, an dem bemessen die Eigenschaften der Elemente eines Moduls beurteilt werden können.

3.1 Lösungsmöglichkeiten der Inhaltsbeschaffung

Die Abschnitte 3.1.1 bis 3.1.3 stellen die Möglichkeiten der Inhaltserstellung in Word ausführlich vor. Es gibt jedoch eine weitere Alternative für die Artikelerstellung, die über den Einsatz einer Web-Anwendung erfolgen kann. Eine strukturierte Eingabemaske, die den Autoren im Internet zur Verfügung gestellt wird, kann der Oberfläche von Word ähnlich gestaltet werden. Die Anpassung der Gestaltung ist erforderlich, um die Eingewöhnungsphase für die Nutzer zu verkürzen. Die eingegebenen Daten können gleich als wohlgeformter³⁴ und valider³⁵ XML-Code geordnet in ein angebundenes Datenhaltungssystem gespeichert und direkt weiterverarbeitet werden. Die Inhaltserstellung über eine solche Web-Maske wird in den Abschnitten 3.1.4 und 3.1.5 dargestellt und näher erläutert. Bei einer solchen Inhaltsbeschaffung kann die Verbindung an ein Web Content Management System (WCMS) ermöglicht werden. Auf genauere Aspekte und Problematik bei Einsatz eines WCMS wird in Kapitel 3.5 eingegangen.

³⁴ Ein XML-Dokument ist wohlgeformt (well formed), falls es sämtliche Regeln für XML einhält.

³⁵ Ein XML-Dokument, welches wohlgeformt ist und ein durch eine Grammatik beschriebenes Format einhält, ist als gültig (valid) definiert. Ein wohlgeformtes und valides XML-Dokument kann von Parsern interpretiert werden und somit die XML-Daten auslesen.

3.1.1 Texterstellung in Word

Bei Word handelt es sich um ein sehr weit verbreitetes und daher allgemein bekanntes Textverarbeitungsprogramm. Auf die einzelnen Funktionen und Möglichkeiten zur Texterstellung im Rahmen dieser Anwendung wird hier deshalb nicht näher eingegangen.

3.1.2 Eingabe von Formeln in Word

Die folgenden Optionen zur Erzeugung mathematischer Formeln sind in Word gegeben:

1. über die Eingabe von Unicode-Werten mittels Tastatur
2. über die Funktion ‚Symbol Einfügen‘ (Unicode-Wert möglich)
3. über den Formeleditor (GIF-Generierung)
4. über das Zusatzprogramm MathType (GIF-/MathML-Generierung).

Mit der Standardinstallation von MS Office wird der Formeleditor nicht installiert. Bei dieser Installationsvariante lassen sich Formeln über die Funktion ‚Symbol Einfügen‘ hinzufügen, oder die einzelnen, vollständig in Unicode definierten Sonderzeichen werden über die Tastatur direkt eingegeben. Die Anleitung findet man in der Hilfefunktion von Word unter ‚Eingeben von Unicode-Zeichen über die Tastatur‘. Für diesen Weg ist eine Umrechnungstabelle mit den benötigten Zeichen und deren Hexadezimalwerte nötig.

Die Alternative über die Funktion ‚Symbol Einfügen‘ stellt die zweite Variante dar. Das folgende Vorgehen ist dazu notwendig. Es muss eine Zeichentabelle (s. Abbildung 6) nach dem jeweils gewünschten Zeichen durchsucht werden; bei der Auswahl ist zu beachten, dass der Zeichencode³⁶ auf der Unicode- und nicht auf dem ASCII-Zeichensatz basiert, da in ASCII nicht alle erforderlichen Formelzeichen enthalten sind. Das benötigte Zeichen ist schließlich zu selektieren und einzufügen. Diese Iteration ist für die einzelnen, nicht vordefinierten Symbole jedes Mal zu wiederholen. Dieser Vorgang nimmt relativ viel Zeit in Anspruch, da jedes einzelne Zeichen der Formel separat per Mausclick in den Text eingefügt werden muss.

³⁶ Evtl. muss in der Zeichentabelle von ASCII auf Unicode umgestellt werden.

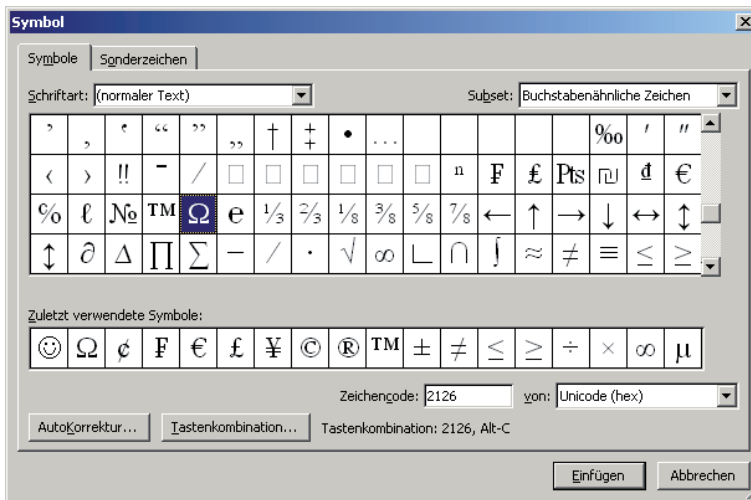


Abb. 6: Screenshot der Zeichentabelle der Funktion ‚Symbol einfügen‘ in Word

Die dritte Variante, der MS Formeleditor, ist zwar in Word integrierbar, muss aber separat über das MS Office-Installationspaket installiert werden. Ist dieser installiert, kann er über die Funktion ‚Objekt Einfügen – MS Formeleditor‘ gestartet werden. Eine Symbolleiste (s. Abbildung 7) wird aufgerufen, die durch eine Abbildung von gängigen mathematischen Zeichen das Eingeben der gewünschten Formel einfach und schnell ermöglicht. Ist eine Formel erst einmal in den Text eingefügt, lässt sie sich problemlos per Doppelclick über die Symbolleiste bearbeiten.

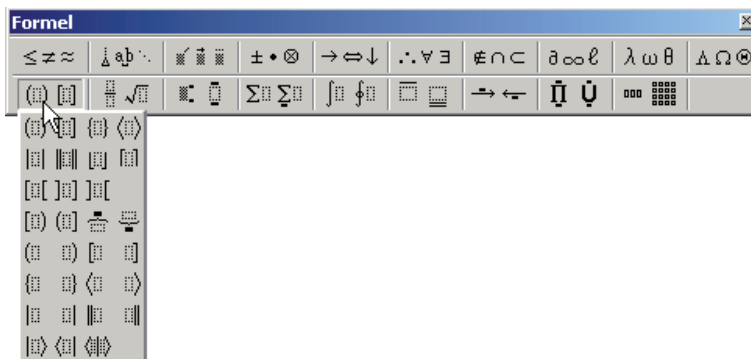


Abb. 7: Screenshot der Symbolleiste des MS Formeleditor

Die Formelerstellung über das Zusatzprogramm MathType funktioniert vergleichbar zum MS Formeleditor. Dieses Programm der Firma DesignScience befindet sich in der Auflistung von MathML-fähigen Formeleditoren, auf die das W3C-Gremium verweist.³⁷ MathType wird dort als professioneller Formeleditor beschrieben. Ist MathType installiert, wird bei Aufruf des Formeleditors in Word automatisch MathType als PlugIn gestartet. In diesem Fall öffnet sich ein neues Fenster mit einem Eingabefeld für die Formeln und einer Tabelle mit zahlreichen Symbolen (s. Abbildung 8). Die Übersichtlichkeit der Zeichentabelle durch eine Gliederung in mathematische Rubriken wie Algebra, Statistik, Geometrie etc. ermöglicht das schnelle Auffinden der

³⁷ Vgl. W3C (2006)

gewünschten Formelzeichen. Zudem bietet MathType nicht nur die Ausgabe in Unicode, sondern auch in MathML an. Zu beachten ist, dass MathType nicht lizenzfrei ist und Anschaffungskosten erfordert.³⁸

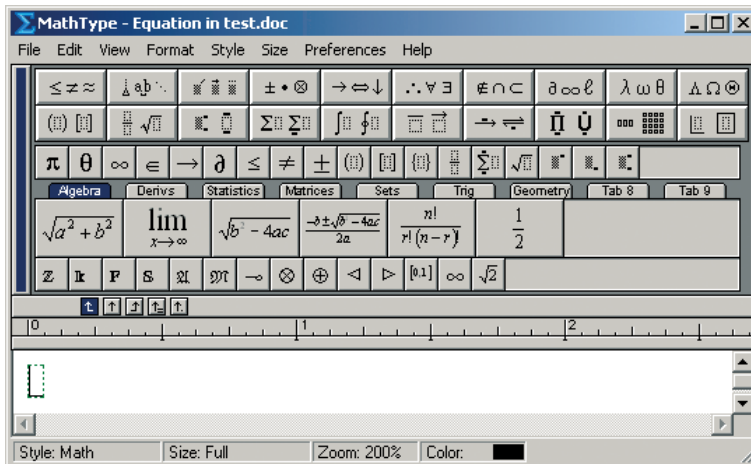


Abb. 8: Screenshot der Anwendung MathType

3.1.3 Integration von Bildern in Word

Die Erstellung von Bildmaterial ist nicht Gegenstand der Arbeit und wird deshalb nachfolgend nicht weiter betrachtet. Es stellt sich aber die Frage, wie diese Art von Inhalten durch die Autoren in einen Artikel integriert bzw. an die DTP-Vorstufe übergeben wird. Für die Integration in Word werden die folgenden drei Möglichkeiten festgelegt:

1. Über die einfache Funktion ‚Grafik aus Datei einfügen‘
2. Über die Funktion ‚Grafik aus Datei einfügen‘ mit Zusatzoption ‚Mit Datei verknüpfen‘
3. Über eine separate Bilddateiübergabe zusätzlich zum Textdokument

Bei der Integration eines Bildes in ein Word-Dokument über die Funktion ‚Grafik aus Datei einfügen‘ wird das Bild in den Word-Code eingebettet; d.h. die Bildinformationen in der Zieldatei (Word-Datei) werden bei einer Änderung der Quelldatei (ursprüngliche Bilddatei) nicht aktualisiert. Eingebettete Bilder werden somit nach dem Einfügen zum Bestandteil der Word-Datei und sind nach dem Einbetten nicht mehr Bestandteil der Bildursprungsdatei.³⁹

Das Einbetten eignet sich besonders für den Autor, der nur eine Datei übergeben muss, da alle Informationen eines Artikels in dem Word-Dokument enthalten sind. Eine separate Übermittlung der Bilddatei ist somit nicht erforderlich und eine korrekte Bildplatzierung ist ebenfalls gegeben. Da drucktaugliche Bilddateien (TIFF, EPS) deutlich mehr Speicherplatz in Anspruch nehmen,

³⁸ Nach dem Umrechnungskurs vom 16.7.2006 betragen die Anschaffungskosten 76.70 €.

³⁹ Vgl. MS Office Word Hilfe

nimmt durch die Einbettung des Bildes auch die Größe der Word-Datei erheblich zu und erhöht dadurch auch die Ressourcenauslastung der Hardware, wodurch eine Verlangsamung der Textverarbeitung stattfindet. Bei älteren Word-Versionen kann es durch die direkte Bildbearbeitung im Word-Dokument zu Qualitätsverlusten des Bildes führen. Wird beispielsweise in der Word-Version 2000 das eingebettete Bild verändert – etwa in der Größe angepasst – wird das Bild im Word-Code in der neuen Form abgespeichert und die ursprüngliche Bildinformation ist nicht mehr rekonstruierbar. Bei der Version 2003 ist dies nicht mehr der Fall. Bei der Bildbearbeitung bleibt der Quellcode immer der ursprüngliche und nur die Darstellung des Bildes wird verändert.

Über die Zusatzoption ‚Mit Datei verknüpfen‘ kann eine Bilddatei verknüpft werden. Das bedeutet, dass in der Word-Datei lediglich der Speicherort der Bilddatei enthalten ist und nicht die vollständige Bilddatei. Die verknüpften Daten werden in der ursprünglichen Quelldatei gespeichert. Die Angaben in der Zieldatei werden nur dann aktualisiert, wenn die Quelldatei verändert wird. Die Dokumentengröße kann über die Funktion gering gehalten werden. Die Bilddatei muss jedoch zusätzlich zum Textdokument übergeben werden.⁴⁰

Eine dritte Möglichkeit Bildmaterial an die DTP-Vorstufe zu übergeben besteht darin, die Bilder weder einzubetten noch zu verknüpfen, sondern die notwendigen Dateien extern dem Word-Textdokument zur Verfügung zu stellen – beispielsweise per Email oder Download von einem angegebenen Server. Dabei ist jedoch zusätzlich abzuklären, an welcher Textstelle das Bild positioniert werden soll. Diese Möglichkeit stellt somit eine aufwendigere Lösung gegenüber den anderen Ansätzen dar und wird aus diesem Grund in der nachfolgenden Bewertung nicht weiter berücksichtigt. Welche Wege der Formelerstellung und Bildintegration mit Word zum Einsatz gebracht werden sollen, wird in der Bewertung des vierten Kapitels behandelt.

3.1.4 Texterstellung und Integration von Bildern über eine Web-Maske

Bei der Inhaltserstellung über eine Web-Maske wird ein XML-Editor verwendet, der dem Autor in Form einer geführten und unterstützten Eingabe über das Internet zur Verfügung gestellt wird. Die Autoren können die Texte bequem über ihren Browser eingeben und bearbeiten. Dabei ist die Maske so aufgebaut, dass die Inhalte schon bei der Eingabe strukturiert werden. Beispielsweise wird der Titel in ein vordefiniertes Feld geschrieben, der Autor in ein anderes usw. Der Text kann über rudimentäre Funktionen bearbeitet werden, so dass beispielsweise Hervorhebungen von Textpassagen durch den Autor möglich sind. Bilder sollen auf den Webserver hochgeladen und abgespeichert werden können. Eine explizite Dokumentenübergabe ist vom Autor nicht erforderlich.

⁴⁰ Vgl. MS Office Word Hilfe

3.1.5 Eingabe von Formeln über eine Web-Maske

Bei der Texterstellung durch Word spielt die effiziente Eingabemöglichkeit von Formeln eine wichtige Rolle. Es stellt sich nun die Frage, wie diese Formelzeichen über einen Webbrowser eingegeben und korrekt dargestellt werden können.

Formeln lassen sich durch folgende Möglichkeiten im Web darstellen:

- 1a. über einen Web-geeigneten Schriftsatz für Sonderzeichen wie die Schriftart ‚Symbol‘⁴¹
- 1b. über die direkte Unicode-Adressierung
2. über Rastergrafiken, d.h. grafische Darstellung in den Bildformaten GIF, JPEG, PNG
3. über die Auszeichnungssprache ‚MathML‘
4. über die skalierbare Vektorgrafik SVG⁴².

Die Formeleingabe bzw. -ausgabe durch die Schriftart Symbol wird als Lösungsmöglichkeit in der Arbeit unter dem Aspekt der Unicode-Adressierung zusammengefasst. Diese erfolgt ebenfalls über einen Schriftsatz (Unicode-basiert), weist somit dieselben Eigenschaften auf und kann daher in dieser Arbeit mit der Formeldarstellung über die Schriftart Symbol gleichgesetzt werden.

Es ist also eine Eingabelösung gesucht, die als Ausgabe Unicode, Rastergrafiken, MathML oder SVG erzeugt. MathML hat durch die Unterscheidung zwischen Syntax⁴³ und Semantik⁴⁴ die größte Verwertbarkeit im Web Workflow. Es kann also ein Wert weitergegeben werden und nicht nur die Darstellungsform der Formel. Allerdings ist die Eingabe für den Nutzer nur über entsprechende Software anwenderfreundlich; die Sicht auf den reinen Quellcode und dessen manuelle Bearbeitung sind keineswegs arbeitsfreundlich und erfordern Fachwissen und Einarbeitungszeit. In der Browser-Darstellung wird MathML nicht von allen Browsern unterstützt – ebenso wenig wie das SVG-Format – und bedarf eines speziellen Browser-PlugIns.⁴⁵ Über SVG und Rastergrafiken lassen sich Formeln auf konventionelle Weise darstellen, sind aber nicht für weitere Berechnungen verwertbar.⁴⁶ In der nachfolgenden Tabelle werden die Vor- und Nachteile nochmals zusammengefasst.

⁴¹ Unter Web-geeigneten Schriftsätzen werden die Systemschriftarten verstanden, die auf jedem Rechnersystem standardmäßig vorhanden sind.

⁴² SVG, Scalable Vector Graphics, ist ein seit 2001 vom W3C-Gremium empfohlener Standard zur Beschreibung zweidimensionaler Vektorgrafiken in der XML-Syntax. SVG ist nicht mit den Bildformaten (GIF, JPEG), die sich bisher im Web etabliert haben gleichzusetzen, da es vektorbasiert ist und in XML beschrieben werden kann.

⁴³ Die Syntax drückt sich durch die korrekte Darstellung im Sinne der mathematischen Schreibweise aus.

⁴⁴ Über die Semantik ist die Software in der Lage, die Form korrekt als Funktion zu interpretieren.

⁴⁵ Auf den Internetseiten des W3C-Gremiums werden Angaben zu unterstützenden Browser und PlugIn gegeben.

⁴⁶ Vgl. Sandhu (2003), S. 7

Formelkodierung	Vorteile	Nachteile
MathML	<ul style="list-style-type: none"> – Formelgehalt für math. Rechnung verwertbar – Formeln konventionell darstellbar: $\frac{a+b}{b} \sqrt{a+b}$ – Formelzeichen mit Text skalierbar – Formel auf Textgrundlinie positionierbar 	<ul style="list-style-type: none"> – Nicht von allen Browsern unterstützt (evtl. PlugIn notwendig) – Kein manuelles Kodieren, Zusatzprogramm (Lizenz) zur Erstellung notwendig – Nicht problemlos in Print Workflow verwertbar
Unicode	<ul style="list-style-type: none"> – Einfach manuelle Eingabemöglichkeit – Formelzeichen mit Text skalierbar – Formel auf Textgrundlinie positionierbar – Von allen Browsern unterstützt 	<ul style="list-style-type: none"> – Formeln nicht für math. Rechnung verwertbar – Nur reduzierte Darstellungsweise möglich: $((a+b) / b) \surd(a+b)$
SVG	<ul style="list-style-type: none"> – Formeln konventionell darstellbar: $\frac{a+b}{b} \sqrt{a+b}$ – Formelzeichen mit Text skalierbar – Formel auf Textgrundlinie positionierbar 	<ul style="list-style-type: none"> – Erstellung von SVG nur über externen Editor oder Konverter möglich – Formeln nicht für math. Rechnung verwertbar – Nicht von allen Browsern unterstützt (PlugIn notwendig) – Nicht problemlos in Print Workflow verwertbar
Rastergrafik	<ul style="list-style-type: none"> – Formeln konventionell darstellbar: $\frac{a+b}{b} \sqrt{a+b}$ – Problemlos in Print- und Web Workflow verwertbar – Von allen Browsern unterstützt 	<ul style="list-style-type: none"> – Formelzeichen mit Text nicht skalierbar – Formel auf nicht exakt auf Textgrundlinie positionierbar – Formeln nicht für math. Rechnung verwertbar – Kein manuelles Kodieren, Bildbearbeitungsprogramm/Formeleditor zur Erstellung notwendig

Tabelle 1: Vor- und Nachteile der verschiedenen Optionen der Formelkodierung

Nachdem die Formelausgabe als MathML, Unicode, SVG und Rastergrafik als grundsätzliche technische Möglichkeiten für die Web-Darstellung von Formeln identifiziert wurden, ist nun zu prüfen, welche Eingabeform zu solch einer Ausgabe führt. Hierzu existieren lediglich zwei Varianten:

1. Über die Integration eines Java-Applet in die Browser-Software. Es gibt Applets, die eine Ausgabe in MathML oder als Rastergrafik (GIF/JPEG/PNG) ermöglichen.
2. Über eine JavaScript-Anwendung, die in den HTML-Quellcode eingebettet wird und eine Ausgabe von Unicode ermöglicht.

Eine Web-basierte Eingabeangewandung von SVG existiert offenkundig bis dato noch nicht. Um SVG-Dateien zu erhalten, besteht die Möglichkeit, MathML-kodierte Formeln über einen Konverter zu SVG umzuwandeln. Dies bietet jedoch keine interessante Alternative, da MathML-kodierte Formeln den Vorteil haben, maschinell verwertbar zu sein. MathML ist daher SVG überlegen; es ist nicht effizient aus einem hochwertigeren Format ein geringeres zu bilden. SVG kann in der Zukunft sicherlich eine interessante Option der Formeldarstellung sein, ohne geeignete Eingabeangewandung ist es das jedoch heute noch nicht.

Eine sinnvolle Eingabemöglichkeit besteht darin, dem Autor des Beitrags eine Zeichentabelle zur Verfügung zu stellen, über die er dem Text die von ihm gewünschten Formelzeichen hinzufügen kann. Entweder geschieht dies über ein Java-Applet, das mit dem Aufruf der Web-Maske in den Browser geladen wird, eine Symbolleiste mit Eingabefeld für Formeln bietet und eine Ausgabe in

MathML oder als Rastergrafik bietet.⁴⁷ Eine solche Lösung kann beispielsweise durch den Einsatz des kommerziell vertriebenen Java-Applets ‚WebEQ Input Control‘ (WebEQ)⁴⁸ der Firma Design Science erfolgen (s. Abbildung 9).

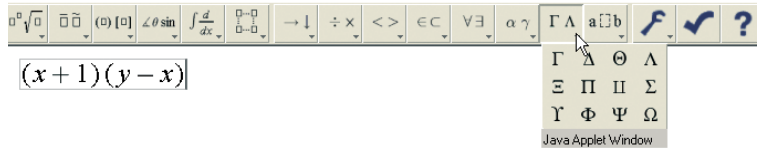


Abb. 9: Beispiel: Java-Applet ‚WebEQ Input Control‘ als Browser-integrierte Symbolleiste zur Formeleingabe

Alternativ ist die Eingabe über eine zusätzliche HTML-Datei, die über einen Link in einem neuen Fenster geöffnet werden kann und über eine grafische Abbildung der Formelzeichen (s. Abbildung 10) per Mausklick die jeweilige Unicode-Adresse der Zeichen anzeigt, möglich. Mittels ‚Copy&Paste‘-Funktion kann dann das Zeichen über die spezifische Unicode-Adresse aus dem geöffneten Fenster in das Textfeld der Web-Maske eingefügt werden.

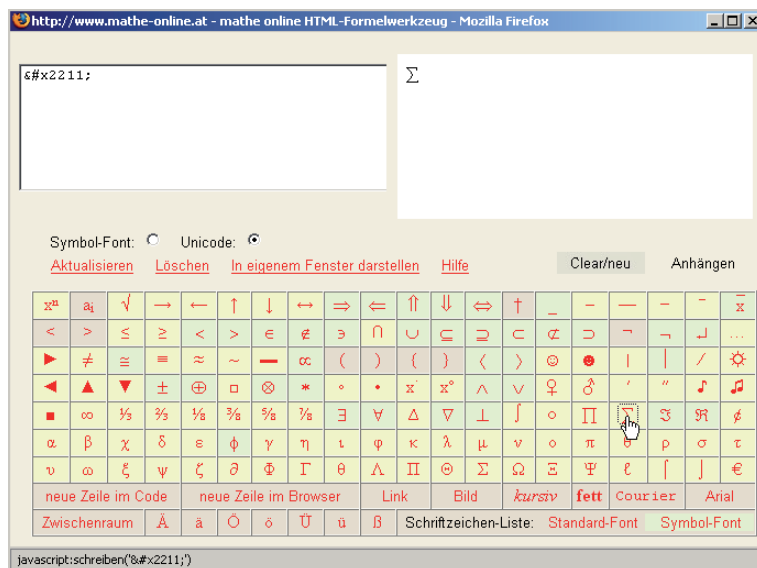


Abb. 10: Mathematisches Formelwerkzeug als JavaScript-Anwendung zur Anzeige der Unicode-Adresse von Formeln, Beispiel⁴⁹

⁴⁷ Um ein längere Laden des Applets zu vermeiden, kann dem Autor auch einmalig das Applet als Download zur Verfügung gestellt werden, das er dann in Form eines Browser-PlugIn auf seinem System installiert und für folgende Inhaltserstellung sofort zur Verfügung steht.

⁴⁸ Die Anwendung ‚WebEQ‘ wird beispielhaft für verschiedene Web-basierte Formeleditoren verwendet. Vgl. dazu Sandhu (2003), S. 360. Eine Liste weiterer Möglichkeiten ist bei W3C (2006) oder in Sandhu (2003), S. 339 ff zu finden.

⁴⁹ Vgl. Mathe Online (2004)

3.2 Lösungswege zur Erzeugung von XML

Ausgehend von den in Kapitel 3.1 dargestellten unterschiedlichen Vorgehensweisen der Inhaltseingabe werden nun die verschiedenen Möglichkeiten zur Erzeugung von XML-Daten betrachtet.

3.2.1 XML aus Word erzeugen

Bei der Inhaltsverarbeitung in Word muss berücksichtigt werden, dass die Autoren verschiedene Word-Versionen zur Erstellung der Inhalte verwenden könnten. Erst in der Version Professional 2003 wird die direkte Ausgabe in XML mittels der Funktion ‚Speichern unter XML‘ unterstützt. Bei früheren Versionen muss auf ein Zusatzprogramm zur XML-Erzeugung zurückgegriffen werden.

Wie bereits erläutert, besteht eine zentrale Anforderung an das Workflow-Konzept darin, dass die Autoren Word zur Inhaltserstellung verwenden müssen. Um die Artikel jedoch vorstrukturieren zu können, kann ihnen eine maßgeschneiderte Dokumenten-Vorlage für Word per Download über das Internet zur Verfügung gestellt werden, an die sie sich zwingend halten müssten. Entscheidend hierbei ist, dass Überschriften, Absätze, Bilder etc. als solche gekennzeichnet werden, damit der Artikel in einzelne Elemente strukturiert und weiterverarbeitet werden kann. Die Vorlage wird anhand einer Strukturdatei (XML-Schema oder DTD) definiert, so dass die in Word gekennzeichneten Absätze und Zeichen problemlos in ein wohlgeformtes XML-Dokument transformiert werden können.

XML aus Word Professional 2003 erzeugen

In Word Professional 2003 lässt sich anhand eines XML-Schemas ein Word-Dokument in ein XML-Dokument konvertieren, das so als Vorlage für die Artikelerstellung dienen kann. Die XML-Technologie muss dabei vom Autor nicht verstanden werden. Allerdings ist sicherzustellen, dass die Regeln des Schemas für die Tags und für die Attribute eingehalten werden, um ein verwertbares XML-Dokument für die Weiterverarbeitung zu erhalten. Diese Bedingung bildet einen kritischen Punkt, da sie die Anforderungen nicht vollständig erfüllt – durch den Einsatz einer XML-Vorlage verändert sich die Oberfläche des Word-Dokuments. Das führt dazu, dass sich der Autor in dieser Ansicht erst neu orientieren muss; anhand einer Dokumentation kann er dann den neuen Ablauf der Artikelerstellung nachvollziehen. Damit das Schema auf ein Dokument angewendet werden kann, ist es notwendig, dass dieses im Programm über ‚Vorlagen/Add-Ins‘ hinzugefügt wurde; der Autor benötigt also nicht nur die Word 2003 Professional-Version, sondern er muss die Kenntnisse besitzen, wie eine Vorlage auf seinem Rechner hinzugefügt wird. Hat der Autor sich in der neuen Dokumentenansicht zurechtgefunden und seinen Artikel unter Berücksichtigung aller Regeln erstellt, kann er selbst ein XML-Dokument erzeugen und an die DTP-Vorstufe übergeben. In der DTP-Vorstufe werden dann alle Artikel gesammelt und zunächst auf die beiden Kriterien ‚Wohlgeformtheit‘ und ‚Validität‘ geprüft, um eine reibungslose Weiterverarbeitung zu Print- und Web-Produkten zu gewährleisten.

XML aus Word über einen Konverter erzeugen

XML-Dokumente lassen sich auch über externe Zusatzprogramme aus Word- bzw. RTF-Dateien erzeugen. Ein solches Zusatzprogramm stellt einen Konverter dar, der Inhalte und Struktur-Anweisungen aus Word-Dokumenten anhand einer DTD oder eines XML-Schemas in ein syntaktisch einwandfreies XML umwandelt. Dieses kann in Form eines PlugIn in die Word-Software integriert oder als eigenständige Anwendung eingesetzt werden.

Bei der Auswahl eines solchen Konverters muss beachtet werden, dass er sowohl MathML, Unicode oder SVG als auch die Verarbeitung von Bildformaten, die verknüpft oder eingebettet sein können, unterstützt.

Der Workflow kann folgendermaßen aussehen: Der Autor erstellt seinen Artikel anhand der bereitgestellten Dokumentenvorlage in Word, speichert diesen als Word- oder RTF-Datei ab und übergibt ihn an die DTP-Vorstufe. Bevor die Daten zur weiteren Verarbeitung an den Layouter oder Webmaster gegeben werden, erzeugt ein Mitarbeiter der DTP-Vorstufe aus der Word- oder RTF-Datei über den Konverter ein XML-Dokument. Die Namensgebung der Strukturelemente muss in der Vorlage, in der Konfiguration des Konverters und in der DTD oder dem XML-Schema aufeinander abgestimmt sein. Der Arbeitsablauf der Autoren wird sich hier nicht verändern. Allerdings entsteht zusätzlicher Personalbedarf, um die vom Autor erhaltenen Dokumente über den Konverter zu XML aufzubereiten. Die Schulung eines Mitarbeiters in der Benutzung und Verwendung des Konverters ist aufwandseitig begrenzt. Fachpersonal ist somit nicht erforderlich. Diese Handlungsmöglichkeit bietet eine gute Option, einen XML-basierten Datenbestand zu erzeugen ohne den gewohnten Arbeitsablauf des Autors zu verändern.

3.2.2 XML über eine Web-Maske erzeugen

Werden die Inhalte über eine Web-Anwendung erstellt, so lassen sie sich direkt medienneutral in ein angebundenes Datenhaltungssystem bzw. im Dateisystem des Webserver ablegen. Das bedeutet, dass Textinhalte in Form von XML-Dokumenten vorliegen, Formeln als MathML, Bildpfad oder Unicode in XML integriert sind und Bilder in einer optionalen Bilddatenbank bzw. auf einem Webserver abgelegt und darüber aufgerufen werden. Eine gemeinsame Datenquelle besteht somit bereits. Voraussetzung dafür ist, dass die Web-Anwendung auf der XML-Technologie aufbaut. Das heißt, bei der Programmierung der Web-Maske muss eine Übergabemöglichkeit der Daten in strukturierter Form berücksichtigt werden. Die Implementierung einer solchen Softwarelösung erfordert speziell qualifiziertes Personal (Web-Entwickler).

3.2.3 XML aus einem Layout-Programm erzeugen

Drittens lässt sich ein XML-Dokument schließlich auch über ein Layout-Programm wie Adobe InDesign CS erzeugen. Das Konvertieren eines InDesign-Dokuments in das XML-Format umfasst

dabei mehrere Schritte: Erstellen oder Laden eines Tag-Satzes, Zuweisung der Tags zu den Absatz- und Zeichenformaten und Export der Struktur und des Inhaltes des InDesign-Dokuments mit dem Befehl ‚Exportieren‘.

XML aus InDesign zu exportieren ist anschaulich in der Hilfe von Adobe InDesign beschrieben und darüber einfach anzuwenden. Zudem können Vorarbeiten für den XML-Export – wie das Erstellen oder Laden eines Tag-Satzes sowie das Zuweisen der Tags zu den Absatzformaten – in einer Vorlage gespeichert werden. Dies birgt den Vorteil, dass diese für wiederkehrende Ausgaben erneut verwendet werden können. Der Aufwand für die Erzeugung einer XML-Ausgabe aus dem Layout-Programm ist gering und zusätzliches Personal ist hierfür ebenfalls nicht erforderlich.

3.3 Integration von nicht text-basierten Inhalten in XML

Der direkte Austausch von XML-strukturierten Textinhalten besitzt den Vorteil, dass diese weder separat übergeben noch aufwendig in XML integriert werden müssen und so direkt verarbeitet werden können. Die Übergabe von Formeln und Bildern ist nach Ausgabeart und Format zu unterscheiden. Die Frage dabei ist, ob sie in XML integriert werden können oder separat behandelt werden müssen. Dies wird im Folgenden näher ausgeführt.

3.3.1 Integration von Formeln in XML

Wie in Kapitel 3.1 beschrieben, existieren bestimmte einfache Wege zur Formeleingabe. Aufgrund der XML-Spezifikation kann die Übergabe von Autorenbeiträgen an die Produktionsvorstufe ebenso einfach erfolgen. Dies steht im deutlichen Kontrast zum Datenaustausch von mathematischen Ausdrücken, welcher sich in den bestehenden Workflow-Konzeptionen noch immer ausgesprochen aufwendig gestaltet. Eine gute, d.h. anforderungsgerechte Lösung steht hierfür noch aus.

2001 wurde zwar vom W3C-Gremium die Auszeichnungssprache MathML als Standard vorgeschlagen, die Eingabe in ein bestimmtes Anwendungsprogramm, die Übergabe zwischen sowie die (Weiter-)Verarbeitung in verschiedenen Arbeitswerkzeugen – charakteristische Merkmale eines Publikationsprozesses – ist aber noch keinesfalls definiert und wird auch nicht von den betrachteten Programmen in der Form unterstützt, dass sich die Schnittstelle zur Ausgabe in Print- und Online-Medien nahtlos gestalten ließe.

Je nach Art der Eingabe und Kodierung werden die Formeln als Bilddatei, Unicode oder MathML in dem erzeugten XML-Dokument verknüpft bzw. dort eingebettet. In Abbildung 11 wird dies grafisch veranschaulicht.

Da der XML-Standard auf dem Unicode-Zeichensatz basiert, ist die Integration von Unicode-basierten Formeln in XML kein Problem – ebenso wenig wie die MathML-Auszeichnung von

Formeln, die sich über einen bestimmten Namensraum in XML beschreiben lässt.⁵⁰ Bei der Kodierung in Form einer Bilddatei, die über den MS Formeleditor ausgegeben werden kann, ist fraglich, was mit den Dateien der Formelbilder passiert, wenn das XML-Dokument übergeben wird. Die Problematik der Bildintegration in XML gehört inhaltlich zu dem folgenden Abschnitt 3.3.2 und wird dort ausführlich behandelt. Könnten die Formeln in dem Vektorgrafikformat SVG ausgegeben werden, so ließen sie sich in XML über einen entsprechenden Namensraum einbinden und übergeben. Es wurde jedoch keine Eingabemöglichkeit gefunden, die SVG als Ausgabe ermöglicht.

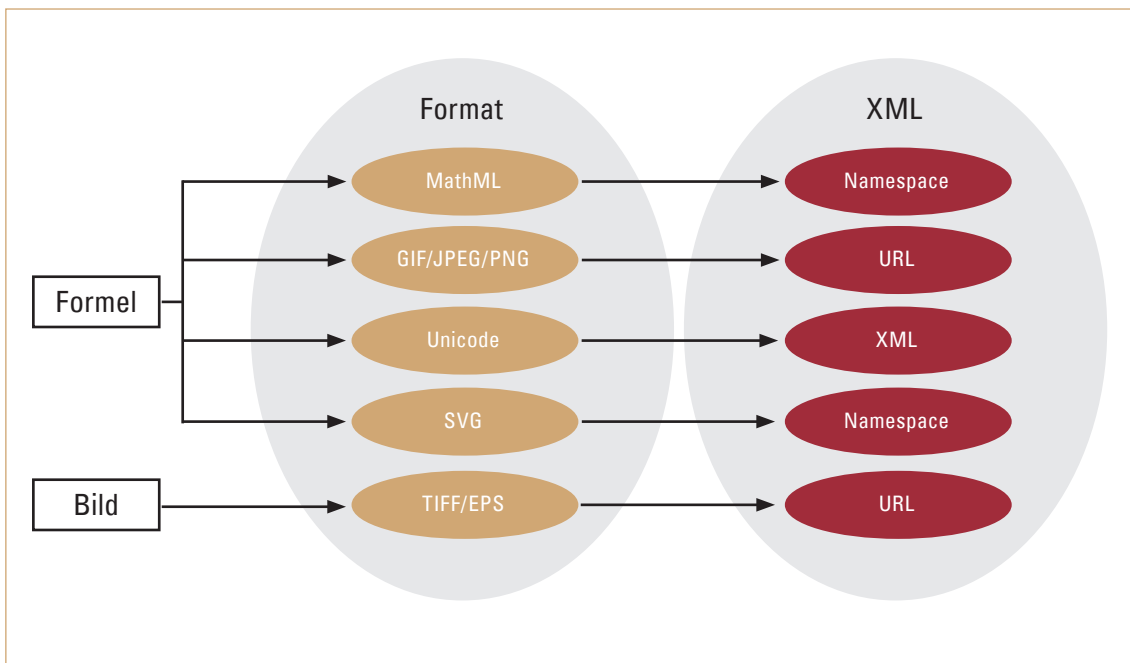


Abb. 11: Schematische Darstellung der Integrationsmöglichkeit von Articlelemente wie Formeln und Bilder in die XML-Auszeichnung

3.3.2 Austausch von Bilddateien über die Integration in XML

Die Übergabe in den verbreiteten Rastergrafikformaten GIF, JPEG und PNG sowie in den drucktauglichen Bildformaten EPS und TIFF kann nicht ohne weiteres über die XML-Spezifikation realisiert werden. Dazu müssten sie in das SVG-Format, das in XML beschrieben werden kann (s. vorherigen Abschnitt), umgewandelt werden. Das SVG-Format ist in der DTP-Vorstufe noch nicht weit verbreitet, da noch die entsprechenden Schnittstellen zu den Layout-Programmen fehlen. Aus diesem Grund wird SVG hier nicht weiter berücksichtigt.

Bilder können in das Textdokument eingefügt und über die Formatvorlage als solche deklariert werden. Bei der Umwandlung in das XML-Format wird das Bild mit seinem Namen bzw. über seine Pfadangabe (URL) als Bild-Tag übergeben (s. Abbildung 11). Das Bild selbst wird aber nicht

⁵⁰ Vgl. zu Namensraum: Sandhu (2003), S. 31 und zu XML und Unicode: Sandhu (2003), S. 32 f

in XML übergeben, sondern über seinen Speicherpfad mit dem XML-Dokument verknüpft. Um eine reibungslose Übergabe des Dokuments inklusive beständiger Pfadangabe zu gewährleisten, muss das Bild dementsprechend geordnet abgelegt werden.

Dabei sind zwei Fälle zu unterscheiden. Enthält die Pfadangabe eine relative Verknüpfung zur Bilddatei, muss diese anschließend zusammen mit der XML-Datei übertragen und gespeichert werden, damit das Bild auch weiterhin verfügbar ist. Wird die Bilddatei jedoch über eine absolute Verknüpfung in die XML-Datei integriert, so muss eine permanente Verfügbarkeit des Bildes garantiert sein. Dies wird über ein Datenhaltungssystem ermöglicht.

3.4 Paralleler versus sequentieller Publikationsprozess

In diesem Kapitel werden die zwei Möglichkeiten der Ablaufreihenfolge des Publikationsprozesses unterschieden und gegenübergestellt. Auf der einen Seite steht der parallel ablaufende Print- und Web Workflow, der von einem medienneutralen Datenbestand ausgeht und drei der vier möglichen Lösungswege widerspiegelt. Auf der anderen Seite laufen Print- und Web Workflow sequentiell ab; dieser Ablauf zeigt den vierten Lösungsweg auf. Zur Veranschaulichung sind die vier benannten Lösungswege in Abbildung 12 grafisch dargestellt.

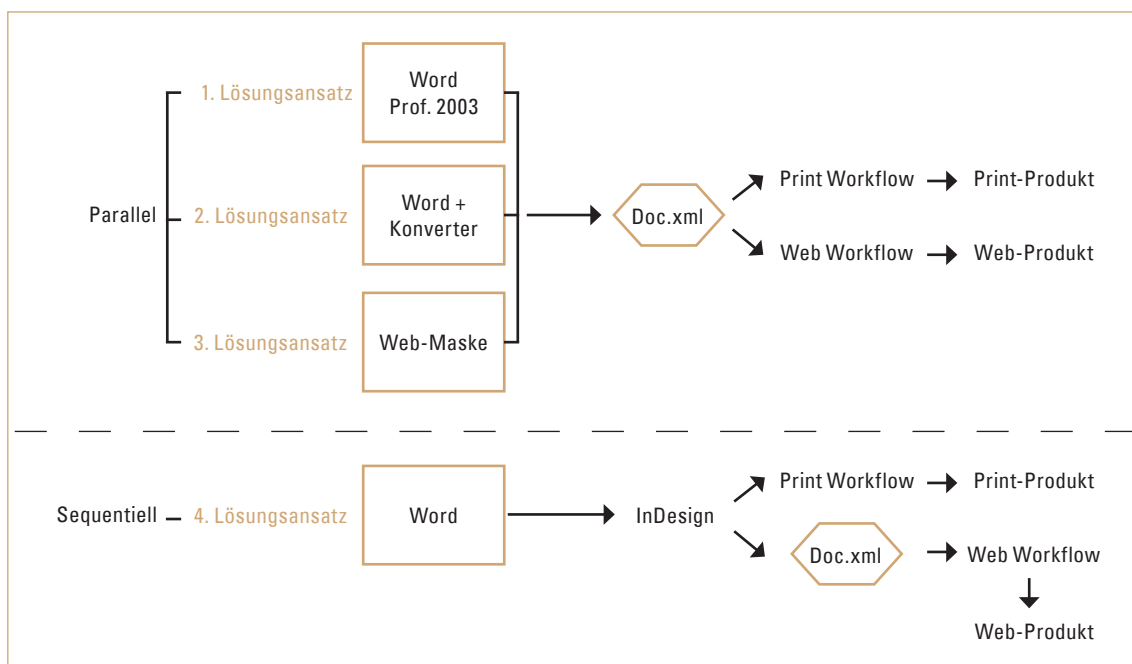


Abb.12: Ablaufreihenfolge schematisch dargestellt: Parallel und sequentiell ablaufenden Lösungswege

Ein Lösungsweg beschreibt eine von vier Vorgehensoptionen des Prozessablaufes und setzt sich aus jeweils einem Element der verschiedenen Module zusammen. Die Möglichkeiten zur Formelerstellung und Bildintegration werden hier noch nicht festgelegt, da die Anwendung eines bestmöglichen Einsatzes erst in der Bewertung im zweiten Teil der Evaluation ermittelt werden muss.

Lösungsansatz 1 beschreibt die Inhaltserstellung und XML-Erzeugung über Word Professional 2003. Die Verarbeitung zu den Endprodukten verläuft parallel. In dem Lösungsansatz 2 werden die Inhalte ebenso über Word erstellt, jedoch über ein Zusatzprogramm in XML konvertiert. Im Anschluss daran läuft wieder ein paralleler Weiterverarbeitungsprozess ab. Lösungsansatz 3 erfolgt über den Einsatz einer Web-Maske. Die Inhalte werden über diese Maske erstellt und direkt in XML strukturiert übergeben. Diese Lösungsmöglichkeit vervollständigt die Vorgehensweisen eines parallelen Prozessablaufes. Lösungsansatz 4 läuft nach dem sequentiellen und somit traditionellen Publikationsprinzip ab. Inhalte werden in Word erstellt, in Form eines Word-Dokuments übergeben, in das Layout-Programm InDesign importiert und zum Druckprodukt verarbeitet. Mit der Druckfreigabe werden über die XML-Schnittstelle des Layout-Programms XML-Daten erzeugt und zum Web-Produkt aufbereitet.

Bei der Betrachtung des sequentiellen Lösungsweges wird deutlich, dass dieser nicht von einem medienneutralen Datenbestand ausgeht. Medienneutralität wurde zwar in den Anforderungen als Effizienzkriterium deklariert, ein sequentieller Ablauf kann aber dennoch eine sinnvolle Lösung darstellen. Dies kann aus betriebswirtschaftlicher und organisatorischer Sicht begründet werden, da die Aufbereitung zu einem strukturierten Datenbestand erheblich mehr Planung und Aufwand durch die Veränderung der Vorgehensweise voraussetzt als bei einem traditionellen Vorgehen.⁵¹

3.4.1 Cross-medialer Publikationsprozess – Single-Source Publishing

Der cross-mediale Publikationsprozess ist hier gleichbedeutend mit der parallel verlaufenden Vorgehensweise. Nach dem Prinzip des so genannten Single Source Publishing werden die Inhalte zu einem Datenbestand aufbereitet, aus dem die Daten parallel für die Druck- und Web-Vorstufe weiterverarbeitet werden können (s. Strukturierung in Abbildung 13). Folglich wird von Single-Source Publishing gesprochen, wenn aus einer einzigen Quelle veröffentlicht wird. Dazu werden XML-Dokumente erzeugt, Formeln werden entweder auf Basis von MathML, Unicode oder Bildform integriert und das entsprechende Bildmaterial wird in einem Datenhaltungssystem abgespeichert und über eine Pfadangabe im XML-Dokument verknüpft. Die Textkontrolle wird mit der Weitergabe an die DTP-Vorstufe abgeschlossen (s. Textfreigabe in Abbildung 13). Es folgen ausschließlich Layout-, Druck- und Web-Freigaben.

⁵¹ Vgl. dazu Rothfuss/Ried (2003), S. 90 f und S. 138

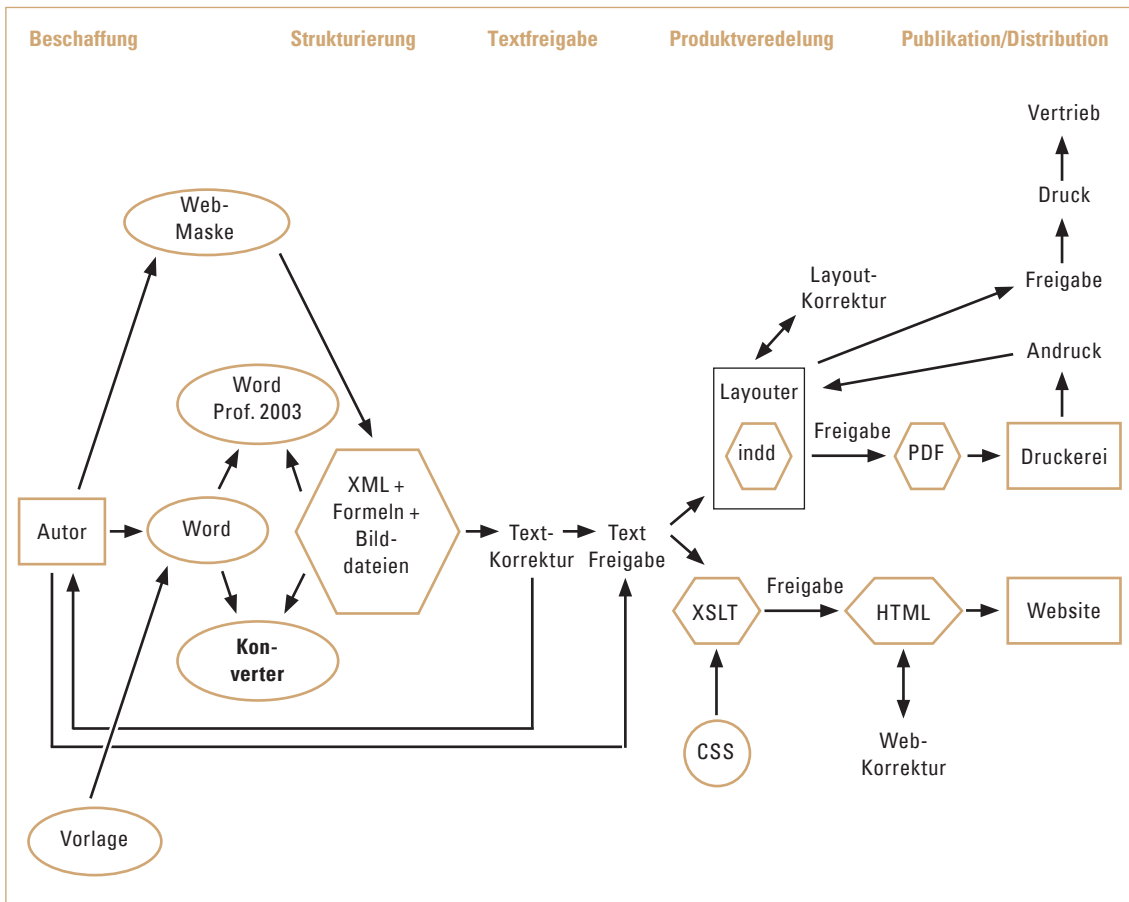


Abb. 13: Schematische Darstellung des parallelen Workflow

Durch die Schaffung eines gemeinsamen Datenbestandes kann eine effektive Arbeitsteilung durch parallele Datenverarbeitung (s. Phase der Produktveredelung in Abbildung 13) erfolgen, Inkonsistenzen zwischen Datenformaten können vermieden werden.

Print Workflow – Verarbeitung im Layout-Programm

Die professionellen DTP-Programme besitzen schon seit längerem XML-Schnittstellen; so auch das hier eingesetzte Layout-Programm ‚InDesign CS 1‘. Die Weiterverarbeitung der XML-Dokumente erfolgt über die Funktion ‚XML importieren‘. Die XML-Elemente werden dann mittels Drag&Drop-Funktion im Dokument positioniert.⁵² Ein reibungsloser Import von XML-Inhalten nach InDesign wird durch eine einheitliche Benennung der Datenstruktur in der XML- sowie der InDesign-Datei ermöglicht. Dies erfolgt über eine DTD, die sich ebenfalls in InDesign importieren lässt. InDesign enthält einen XML-Parser, der Tags anhand dieser DTD-Datei validieren kann.

⁵² Genauere Erklärungen zum Import von XML-Dateien in InDesign CS können in der Adobe InDesign-Hilfe unter „Arbeiten mit XML-Dateien“ gefunden werden.

Formeln, die über MathML ausgezeichnet und in das XML-Dokument integriert werden, können über die Import-Funktion dem Layout-Dokument hinzugefügt werden. Dabei werden aber die Formeln nicht in mathematischer Darstellungsweise angezeigt, sondern nur als funktionaler, mathematischer Zusammenhang in Textform – und sind somit für den Leser nicht als mathematischer Ausdruck zu erkennen. Um diesen in der für den Anwender gewohnten Darstellungsform wiederzugeben, muss er in einem Anwendungsprogramm – ob Layout-Programm oder Webbrowser – über ein Zusatzprogramm interpretiert und anschaulich gemacht werden. Allerdings existiert für das Layout-Programm Indesign CS bis dato kein solches PlugIn.

Die unicode-basierte Formelübergabe weist demgegenüber diese Probleme nicht auf, da diese Formeln von Layout-Programmen über eine Unicode-Schriftart erkannt werden können. Die Schriftart muss dann wiederum in der eingesetzten Stilvorlage für Formeln hinterlegt sein, damit die Formeln auch automatisch erkannt und nicht über Platzhalter dargestellt werden.

Im XML-Dokument integrierte Bilder werden beim Importieren über ihren Bildpfad erkannt. InDesign greift automatisch über den angegebenen Pfad auf das Bild zu und – vorausgesetzt der Pfad stimmt – erstellt ein Vorschaubild.

Web-Workflow – Verarbeitung zum Online-Medium

Parallel zum Print Workflow läuft die Verarbeitung zum Online-Medium ab. Der Webmaster kann mit einem beliebigen Web-Editor die notwendigen Artikelelemente, die in XML vorliegen, in vordefinierte Felder von HTML-Vorlagen einlesen und als HTML ausgeben.

Je nach Art der Formel Ausgabe müssen die Formeln für die Web-Ausgabe angepasst werden. Unicodebasierte Formelverarbeitung sollte aufgrund der indirekten Darstellungsweise von Brüchen, Wurzeln etc. über einen Formeleditor neu erstellt und in einer anderen Kodierungsform (als MathML oder Rastergrafik) ausgegeben werden. Rastergrafiken können in HTML als Bilddatei eingebunden werden, die dabei entstehenden Schwierigkeiten der exakten Positionierung auf der Grundlinie des Textes sollten dabei durch einen neu eingefügten Absatz, der ausschließlich der Formelwiedergabe dient, umgangen werden. Für die Darstellung MathML-ausgezeichneter Formeln muss diese im HTML-Quellcode eingebettet werden. Die entsprechenden Elemente müssen dann über einen Interpreter im Browser darstellbar gemacht werden. Das hoch aufgelöst vorliegende Bildmaterial der Artikel kann über einen Konverter Web-gerecht umgewandelt werden.

Anstatt die Inhalte vom Webmaster manuell in HTML-Vorlagen einzubinden, lässt sich bei einem medienneutralen Datenbestand die Erstellung des Web-Produkts auch automatisieren. Die vorhandenen XML-Daten und die dazugehörigen Bilddateien werden über einen Transformationsprozess in eine komplette Website umgewandelt. Dafür wird ein Transformationsprozessor eingesetzt, welcher die Aufgabe hat, die XML-Elemente in HTML-Elemente umzuwandeln. Die

Steueranweisungen für solch einen Prozessor, die bei der Transformation benötigt werden, werden in einer externen Datei beschrieben. Dies kann über die Extensible Stylesheet Language Transformations (XSLT) realisiert werden, die in Form von Regeln bestimmt, wie Quelldokumente (XML) in ein gewünschtes Zielformat (HTML) umgewandelt werden sollen. Über eine mögliche Batch-Programmierung kann dieser Prozess automatisch angestoßen werden, so dass ein aktives Eingreifen nicht erforderlich ist. Dadurch wird die Integration des Web Workflow vollständig automatisiert und entspricht den genannten Anforderungen. Der Aufwand ist dabei hoch, fällt aber nur einmalig an.

3.4.2 Sequentiell verlaufender Publikationsprozess – Erst Print dann Web

Die sequentielle Abfolge eines Workflow setzt voraus, dass zunächst die Daten für das Druckprodukt vollständig erzeugt, korrigiert und freigegeben sein müssen, um anschließend die Datenaufbereitung für das Web vornehmen zu können. In Abbildung 14 ist diese Abfolge grafisch dargestellt; auf die Artikelerstellung folgt die Veredelung des Print-Produkts, dann dessen Druckfreigabe und anschließend die Veredelung des Web-Mediums.

Bei dieser Ausgestaltung des Workflow wird ein neuer Lösungsansatz erst beim Layouter eingebracht. Die Korrekturschritte erfolgen primär im Layout-Programm, indem das vom Layouter erstellte PDF-Dokument an die Autoren zur inhaltlichen Korrektur von Text- und Bildmaterial zurückgesandt wird. Der inhaltliche Datenbestand ist folglich erst nach Abschluss dieser Prozessphase fehlerbereinigt und durch die Autoren freigegeben; die Aufbereitung des Web-Produkts ist auch dann erst sinnvoll.

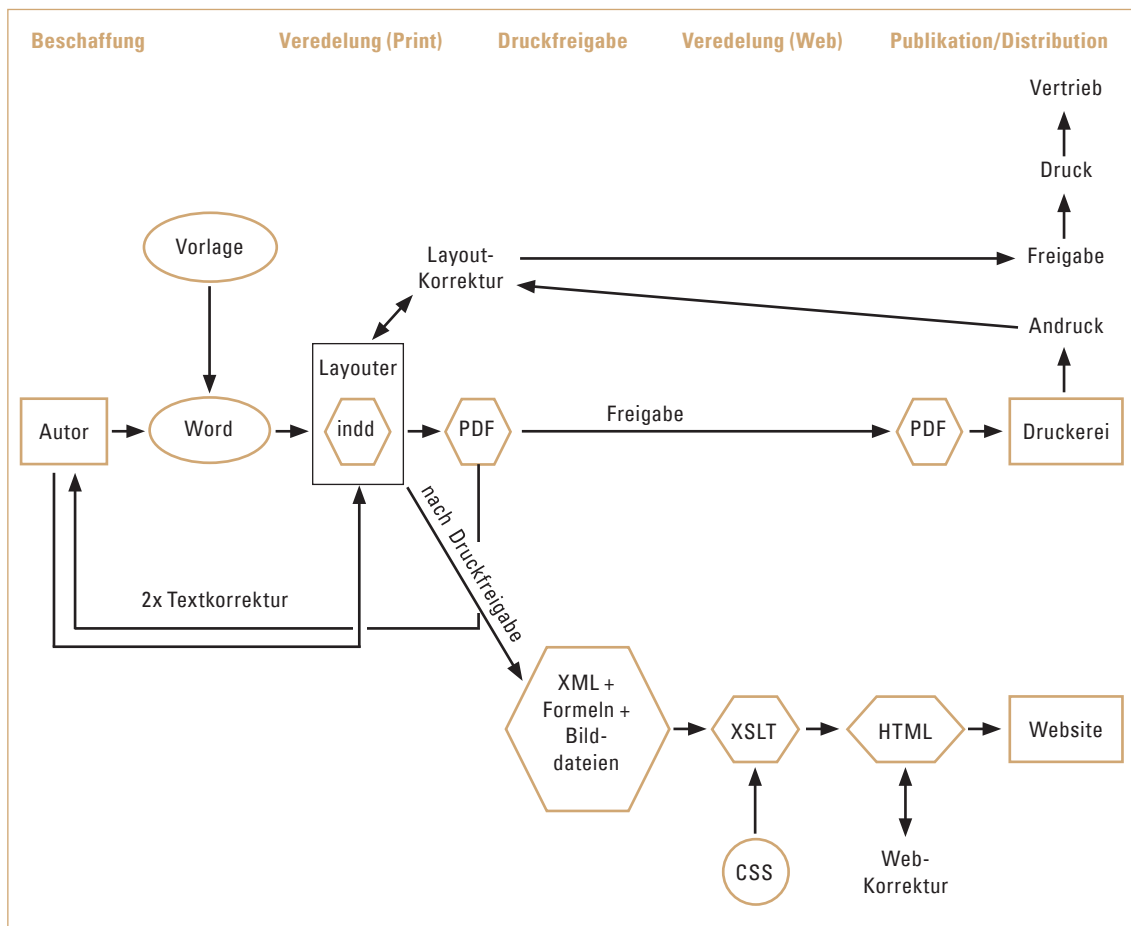


Abb. 14: Schematische Darstellung des sequentiellen Workflow

Der Workflow enthält somit grundsätzlich kein neues Gestaltungskonzept, erfährt allerdings zahlreiche Verbesserungen. Durch die Möglichkeit, über die XML-Schnittstelle des Layout-Programms die freigegebenen Druckdaten in XML-Daten umzuwandeln, kann das Web-Produkt automatisiert erstellt werden und auch den hier geforderten Anforderungen entsprechen.

Print Workflow – Verarbeitung im Layout-Programm

Um die Textbearbeitung für den Layouter zu vereinfachen, ist dringend zu empfehlen, dass die Autoren eine bereitgestellte Formatvorlage für ihre Articlelemente verwenden. Ist das Format der Vorlage für Word äquivalent zu dem Format der Stilvorlage in InDesign definiert, so kann das Word-Format nach dem Import vom InDesign-Format unmittelbar erkannt und ggf. angepasst werden. Dies stellt eine erste mögliche Verbesserung im Vergleich zur Ausgangslage dar, in welcher die Word-Texte mittels Copy&Paste in das Layout-Dokument eingefügt werden, dadurch keine Formatierung erkannt wird und diese folglich noch hinzugefügt werden muss. Ein text-basierter Workflow über die Import-Funktion ist also kein Problem – was mit den Formeln und Bildern geschieht, wird im Folgenden diskutiert.

Durch den Import der Word-Dokumente in das Layout-Programm werden die Formeln entweder im TIFF-Format oder als Unicode-Zeichen – je nach Eingabeart – übernommen.⁵³ Die TIFF-Datei wird beim Import in InDesign neu generiert und in einen standardmäßig festgelegten Dateiodner auf der lokalen Festplatte beim Anwender abgelegt. Die Import-Option ‚Formatierung beibehalten‘ ist allerdings zu aktivieren, damit die Formel korrekt dargestellt wird. Wird diese Option ignoriert, wird kein TIFF-Format erzeugt; eine Unicode-Formel wird in diesem Fall auch nur teilweise korrekt gezeigt, da keine Angaben zur Formatierung übergeben werden können. Zudem ist die Qualität des erzeugten TIFF-Bildes zu gering, um drucktauglich zu sein. Somit lässt sich feststellen, dass beide Varianten keine zufrieden stellende Lösung für die Qualitätsanforderungen an das Druckprodukt bieten. Die Formeln müssen daher über ein Zusatzprogramm⁵⁴ nachgebaut und als drucktaugliches Bild importiert werden. Im Fall einer unicodebasierten Ausgabe sind die Formeln über eine Schriftart wie ‚Mathematical Pi 1‘, ‚Times New Roman‘ oder ‚Arial Unicode MS‘, die alle Zeichen beinhalten, unter Verwendung verankerter Rahmen darzustellen. Für eine korrekte Eingabe und Darstellung von Formeln in einem Layout-Programm muss also eine sinnvolle Lösung gefunden werden. Speziell für die Layout-Programm Indesign wurde ab der Produktversion CS 1 ein PlugIn⁵⁵ entwickelt, welches dieses Problem lösen soll. Nach weiterer Entwicklung soll darüber auch in Zukunft die Möglichkeit gegeben werden, Formeln in MathML direkt auszugeben, was für den gesamten Workflow von großer Bedeutung wäre.

Bilder, die in den Word-Code eingebettet wurden, können beim Import automatisch übernommen und angezeigt werden. Das Bild wird hierbei in das TIFF-Format konvertiert und separat auf der lokalen Festplatte abgelegt.⁵⁶ Bilder, die hingegen mit der Word-Datei verknüpft werden, werden beim Import nicht erkannt und nur als leere Bildrahmen angezeigt.

Für den automatischen Import ist es also zwingend erforderlich, dass Bilddateien eingebettet werden – allerdings kann dies die Bildqualität beeinträchtigen. Eine Variante des Vorgehens besteht darin, die Bilder bei separat vorliegenden Bilddateien einzeln zu importieren und manuell im Layout zu positionieren. Dies erfordert jedoch den zusätzlichen Aufwand, die exakte Bildposition aus dem Word-Dokument zu ermitteln.

XML-Einsatz in der Verarbeitung zum Web-Produkt

XML wird mit der Fertigstellung des Druckprodukts über die Export-Funktion im Layout-Programm eingesetzt. In InDesign werden beim Export von Dokumenten in XML nur Elemente mit Tags ein-

⁵³ Vgl. Kapitel 3.4.1, in dem die Übergabe von unicodebasierten Formeln in ein Layout-Programm erklärt wird. Auch wenn die Formeleingabe über den Formeleditor MathType, der die Ausgabe in MathML ermöglicht, erfolgt, wird beim Import des Word-Dokuments in das Layout-Programm automatisch ein Bild generiert und übergeben.

⁵⁴ Hier bieten sich Programme wie Illustrator von Adobe oder auch eine eigenständig laufende Anwendung eines Formeleditors wie MathType an.

⁵⁵ Dieses PlugIn nennt sich ‚InMath‘ und wurde von der Firma i.t.i.p. GmbH entwickelt; es handelt sich um einen PlugIn für echten mathematischen Formelsatz.

⁵⁶ Vgl. dazu den vorherigen Abschnitt zur Integration und Übergabe von Formeln in Bildformat.

bezogen. Formatierungen und Layout-Anweisungen gehen verloren. Anhand einer importierten DTD werden die zulässigen Strukturen der zu exportierenden XML-Datei definiert. Zuweisungen von XML-Tags zu InDesign-Formaten können in einer Vorlage gespeichert werden und müssen dadurch nur einmal gesetzt werden. Sie können dann für weitere Publikationsausgaben wiederverwendet werden.⁵⁷

Die in einem Text enthaltenen Formeln werden, sofern sie in Bildform vorliegen, wie Bilder behandelt (siehe dazu den nachfolgenden Absatz). Werden sie über Zeichen, also in Unicode, dargestellt, werden sie automatisch als Inhalt in der XML-Auszeichnung integriert und können aus dieser entsprechend exportiert und an andere Anwendungen übergeben werden. MathML-kodierte Formeln können im Layout-Programm bisher nicht verarbeitet und folglich auch nicht über die XML-Schnittstelle ausgegeben werden.

Wie in Kapitel 3.3.2 beschrieben, können Bilder nicht direkt über die XML-Technologie übergeben werden. Im Layout-Dokument enthaltene Bilder sind deshalb auch hier gesondert zu behandeln. Sie können über XML-Tags entsprechend gekennzeichnet werden. InDesign erstellt automatisch ein ‚href‘-Attribut, das den Speicherort der Bilddatei angibt. Dabei sind zwei Fälle zu unterscheiden. Befindet sich die Bilddatei in demselben Verzeichnis wie die zu exportierende XML-Datei, enthält das ‚href‘-Attribut eine relative Verknüpfung. Wird diese anschließend zusammen mit der XML-Datei übertragen, ist das Bild auch weiterhin verfügbar. Befindet sich jedoch die Bilddatei in einem anderen Dateiverzeichnis als die XML-Datei, so beinhaltet das ‚href‘-Attribut eine absolute Verknüpfung, die in übertragenen Dateien sehr wahrscheinlich nicht intakt bleibt.

Beim Exportieren bietet InDesign die Option, Web-gerechte Bilder (GIF, JPEG) zu erstellen. Diese werden automatisch mit dem Originalbild zusammen in einem Unterordner ‚Bilder‘ gespeichert. Die ‚href‘-Attribute enthalten dann die relativen Verknüpfungen zu diesen neu generierten Bildern. Die exportierten Daten können anschließend für die Online-Ausgabe automatisiert weiterverarbeitet werden.

Automatisieren des Web Workflow

Analog zu einem parallel verlaufenden Workflow-Konzept lässt sich die Erstellung des Web-Produkts auch bei einem sequentiellen Ablauf automatisieren. Bedingung hierfür ist ein XML-basierter Datenbestand, der über das Layout-Programm erzeugt werden kann. Die weiteren Schritte gleichen der in Kapitel 3.4.1 beschriebenen Verarbeitung zum Online-Medium. Ebenso würde die Integration des Web Workflow dadurch voll automatisch gewährleistet sein und den genannten Anforderungen entsprechen.

⁵⁷ Eine genauere Beschreibung zu der Frage, wie XML-Formate aus InDesign konvertiert werden können, ist in der Hilfe von Adobe InDesign Hilfe unter dem Schlagwort ‚Arbeiten mit XML-Dateien‘ zu finden.

3.5 Content Management Systeme als integrativer Lösungsansatz

Denselben Datenbestand für verschiedene Publikationsprodukte zu verwenden stellt mittlerweile ein gebräuchliches Vorgehen in der Publikationserstellung dar. Folglich wird für diesen Zweck auch eine ganze Reihe von entsprechenden Content Management Systemen (CMS) auf dem Markt angeboten. Beispielsweise könnte in dem hier betrachteten Anwendungsfall, bei dem die Inhaltsbeschaffung über einen geografisch weit verteilten Autorenkreis erfolgt, ein Web-basiertes CMS (WCMS) dazu eingesetzt werden, um den in Kapitel 2.1 beschriebenen redaktionellen Workflow in der Beschaffung, aber auch Veredelung, Verwaltung und Wiederverwendung der Daten zu unterstützen. Die primären Entscheidungskriterien für ein solches System sind beispielsweise ein erheblicher Umfang des Webauftritts (d.h. mehr als 1.000 Webseiten), sehr kurze Aktualisierungszeiten oder eine möglichst dezentrale Websiteverwaltung.⁵⁸

Die einzelnen Funktionen solcher Systeme bieten auch für den in dieser Arbeit betrachteten Publikationstyp Vorteile im Umgang mit den Daten; allerdings sind die entsprechenden Anforderungen an die Daten hier deutlich geringer, so dass sich der Aufwand zur Einrichtung und zum Betrieb eines WCMS hierfür nicht lohnt; lediglich für die Abfragefunktion würden sich aufwandseitig gerechtfertigte Vorteile ergeben. Zudem sollen im hier untersuchten Anwendungsfall nicht gewöhnliche Inhalte wie Texte, Bilder und Medienobjekte publiziert werden, sondern wissenschaftliche Texte mit ihren mathematisch-technischen Funktionen, deren Darstellung einen hohen Anspruch an den Workflow erhebt. Ein entsprechendes redaktionelles System muss daher insbesondere die Verarbeitung mathematischer Ausdrücke in Syntax und Semantik ermöglichen. Derartige Systeme sind mit zusätzlich hohen Kosten verbunden.⁵⁹ Die Kosteneinsparungen im Workflow, die über ein solches System also zunächst erzielt werden können, werden durch die beträchtlichen Zusatzkosten für Anschaffung und Implementierung überkompensiert. Des Weiteren erfüllen kleine wissenschaftliche Publikationen auch nicht die oben genannten Kriterien, um die Entscheidung für den Einsatz eines erfolgreichen WCMS zu begründen.

Es wird hier die Auffassung vertreten, dass kleine Publikationen mit komplexen technischen Prozessen, die durch den Einsatz einer Systemlösung geboten wären, meist nicht gut bedient sind. Für die kleine Publikation einer wissenschaftlichen Fachzeitschrift mit dem besonderen Anspruch der Formelverarbeitung gilt diese Meinung erst recht. Die Option ‚CMS‘ scheidet deshalb unter den gegebenen Anforderungen als mögliche Lösung aus; sie wird im weiteren Verlauf nicht mehr verfolgt.⁶⁰

⁵⁸ Vgl. Kretzschmar/Dreyer (2004), S. 197. Eine Aufzählung möglicher Entscheidungskriterien ist auch dem Artikel von Marti (2006) zu entnehmen.

⁵⁹ Aus dem Produktfinder des Online-Fachmagazins ‚Das Content Management Portal‘ sind entsprechende Systemlösungen ab 4.500 € zu finden. Im Fall einer Open Source Lösung ist zu berücksichtigen, dass kein Service für Installation und Wartung angeboten wird und die Kosten durch Personalaufwand erheblich steigen. Vgl. Contentmanager.de (2006)

⁶⁰ Marti schreibt dazu: „Je komplexer und aufwändiger die Prozesse in einem Unternehmen sind, desto sinnvoller ist der Einsatz eines CMP-Systems.“ Zitat aus Marti (2006). Marti bezieht sich hier zwar auf Cross Media Publishing Systeme, die Aussage ist aber grundsätzlich auch auf sonstige Systemlösungen wie WCMS zu übertragen.

3.6 Bestimmung des Kriterienkataloges

Die Kriterien erfassen das Eigenschaftsprofil eines Evaluationsobjektes, gemessen an den Anforderungen und den Zielvorstellungen. Sie werden so ausgewählt, dass unter unterschiedlichen Objekten das am ehesten Geeignete identifiziert werden kann. Sie können deshalb als das Bindeglied zwischen Aufgabenstellung und erfolgreicher Zielumsetzung bezeichnet werden.⁶¹

Im Folgenden werden diese Kriterien erarbeitet und gegliedert. Dabei sind sie weiterhin in die Prozessphasen Formelerstellung, Bildintegration, XML-Erzeugung und Prozessablauf unterteilt, abschließend werden die entstehenden Kosten berücksichtigt. Besonders hilfreich für die Überprüfung erscheint es, die einzelnen Beurteilungskriterien in Frageform zu kleiden. So kann einfacher und deutlicher ermittelt werden, inwieweit das jeweilige Merkmal erfüllt ist.

Es ist hier nochmals zu verdeutlichen, dass jede Prozessphase sowie der Aspekt der Kosten ein eigenes Lösungsmodul darstellt, das sich aus mehreren Lösungselementen zusammensetzt. Für einen kompletten Lösungsweg ist eine Kombination von Lösungselementen erforderlich, die entsprechend aus dem jeweiligen Modul ausgewählt werden. Zu Beginn jedes folgenden Abschnitts wird – soweit es der besseren Übersichtlichkeit dient – anhand einer entsprechenden Abbildung das jeweilige Lösungsmodul (beige) und die enthaltenen Lösungselemente (rot) dargestellt.

Formelerstellung

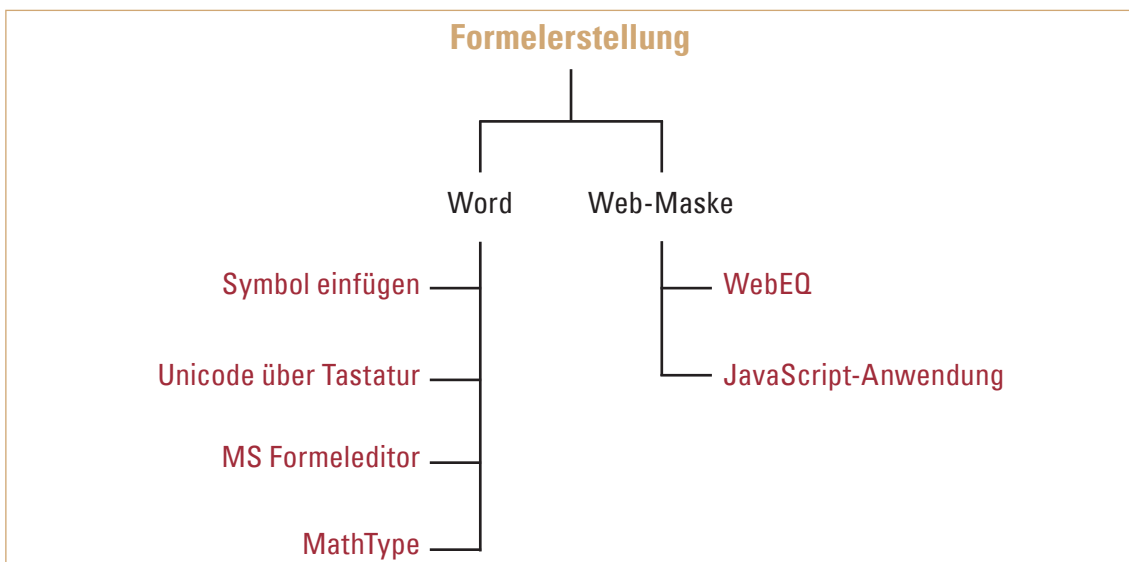


Abb. 15: Lösungsmodul ‚Formelerstellung‘

Die Eingabeapplikationen zur Erstellung von Formeln sind unter den Kriterien der Anwenderfreundlichkeit und des Installations- und Kostenaufwandes auf der Seite der Autoren zu beurteilen.

⁶¹ Vgl. Heinrich/Häntschel (2000), S. 17ff

1. Anwenderfreundlichkeit

- a. Eingabe – Wie anwenderfreundlich ist die Eingabe von Formeln? Erfolgt sie mittels einer Symbolleiste, einer Zeichentabelle oder direkt per Tastatureingabe?
- b. Bearbeitung – Wie lässt sich eine Formel bearbeiten – in einzelnen Zeichen oder als Formel im Ganzen?
- c. Zusatzinformation – Benötigt der Nutzer zusätzliche Informationen, um eine Formel eingeben zu können, wie z.B. Hexadezimalwerte von Unicode-Zeichen?
- d. Dokumentation – Eine Dokumentation beispielsweise in Form einer Hilfe-Funktion unterstützt den Autor bei Problemen der Inhaltserstellung. Ist diese bereits standardmäßig in der Anwendung integriert oder muss sie neu erstellt werden?

2. Software

Installationsaufwand beim Nutzer (Autor) – Muss ein Zusatzprogramm für die Formelerstellung installiert oder eingerichtet werden?

3. Kosten auf Autoreseite

Es fallen Anschaffungskosten für eine Eingabe-Software für Formeln an. Hierbei werden nur die Kosten berücksichtigt, die direkt beim Autor anfallen.⁶²

- a. Lizenzkosten – Ist eine Lizenz für eine Anwendung zu erwerben?
- b. Upgrade-Kosten – Ist ein Upgrade der Produktversion der Anwendung erforderlich?

Formelkodierung

Die Kodierungsarten einer Formel sind separat zur Formeleingabe zu betrachten, da sie einen unterschiedlichen Nutzwert für den anschließenden Verarbeitungsprozess besitzen. Dabei werden folgende Darstellungs- und Verarbeitungs-Kriterien den verschiedenen Formelausgabemöglichkeiten gegenübergestellt:

1. Darstellung

- a. Layout-Darstellung – Werden die Formeln in mathematisch korrekter Darstellungsform abgebildet? Können Bruchstriche, Wurzeln, Klammern etc. sich über eine gesamte Funktion erstrecken?
- b. Formelpositionierung – Kann die Formel auf der Grundlinie des Textes im Layout-Programm oder in der Web-Darstellung exakt positioniert werden?
- c. Formelskalierbarkeit – Können die Formelzeichen mit dem Text vergrößert oder verkleinert werden?
- d. Browser-Darstellung – Ist die Formel in dieser Kodierung problemlos über einen Browser darstellbar oder ist ein PlugIn dafür notwendig?

⁶² Kosten, die auf der Herausgeberseite anfallen, werden später separat über das Modul ‚Kosten‘ berücksichtigt.

2. Verarbeitung

- a. Manuelle Bearbeitung – Ist der Quellcode der Formel verständlich und können daher einzelne Formelbestandteile manuell verändert werden (z.B. ein Bild wird über seinen Quellcode nicht verstanden und benötigt zur Formelbearbeitung die Eingabe-Software)?
- b. Übergabe – Kann die Kodierungsart der Formel in XML integriert und mit dem XML-Dokument übergeben werden?
- c. Konvertierung Print/Web – Ist eine Konvertierung der Formel für die Darstellung im Print- bzw. Web-Produkt notwendig?
- d. Import in Layout-Programm – Wird der Import der Formel in das Layout-Programm unterstützt und wird die Formel somit korrekt übergeben und dargestellt?
- e. Export aus Layout-Programm – Kann der vorhandene Ausdruck in dieser Kodierungsart über die (interne) XML-Schnittstelle des Layout-Programms exportiert und in einem XML-Dokument integriert werden?
- f. Verwertbarkeit Web – Kann der mathematisch-technische Ausdruck unmittelbar durch andere computergestützte Anwendungen für weitere Berechnungen genutzt werden?

Bildintegration

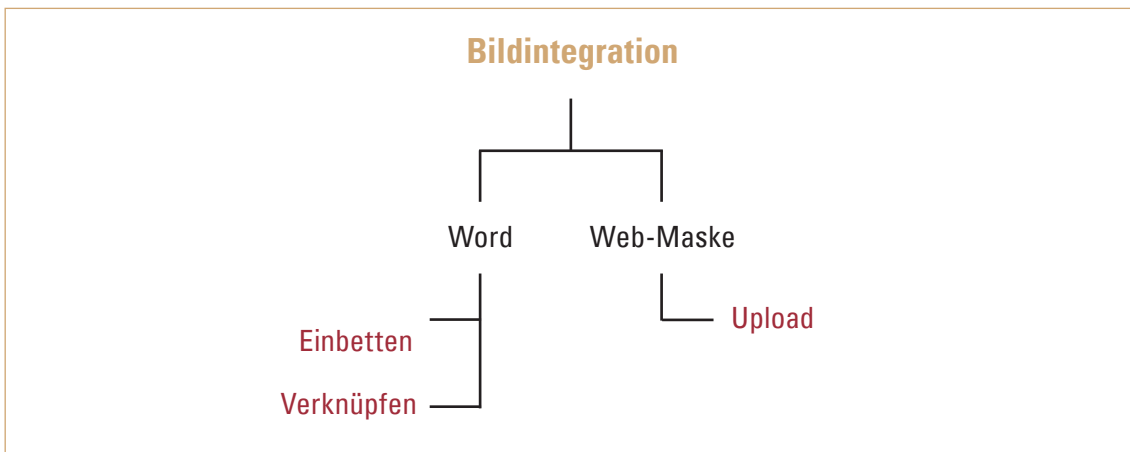


Abb. 16: Lösungsmodul ‚Bildintegration‘

Bilder und Grafiken werden in Word standardmäßig unterschiedlich integriert.⁶³ Unabhängig von den Produktversionen können jedoch zwei Möglichkeiten zusammengefasst werden: Einbettung oder Verknüpfung von Bildern. Zur Bildintegration in Word kommt der Bild-Upload bei eingesetzter Web-Maske in der Bewertung hinzu.

Folgende Kriterien sind dabei zu bewerten:

1. Bildplatzierung – Kann ein Bild im Textfluss positioniert werden bzw. wird deutlich, an welche Stelle innerhalb des Textes ein Bild platziert werden soll?

⁶³ Vgl. Meister/Gahler/Jamieson/Freßdorf (2006), S. 334

2. Dateigröße – Verändert sich die Dateigröße des TextDokuments bei der Integration eines Bildes?
3. Übergabe – Wie wird ein Bild bei der Dokumentübergabe überliefert? Wird es in das Textdokument eingebettet oder muss es separat übergeben werden, da es verknüpft wurde?
4. Speicherort – Wird das Bild automatisch in einem angebenen Datenhaltungssystem abgelegt oder muss es manuell eingepflegt werden?
5. Ursprungsformat – Bleibt die ursprüngliche Bilddatei nach der Bildintegration und der Bildübergabe unverändert erhalten?

XML-Erzeugung

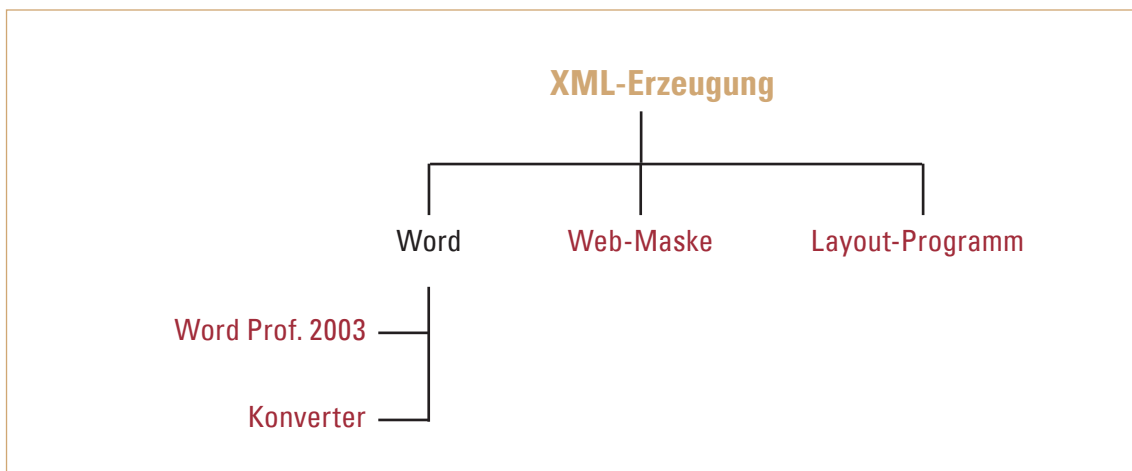


Abb. 17: Lösungsmodul ‚XML-Erzeugung‘

Für die Generierung von XML-Daten existieren vier unterschiedliche Lösungsmöglichkeiten (in Abbildung blau eingefärbt). Diese sind:

1. Die Erzeugung von XML-Daten mit Word, Produktversion ‚Professional 2003‘
2. Die Erzeugung von XML-Daten mit Word in einer älteren Produktversion und zusätzlichem XML-Konverter
3. Die (indirekte) Erzeugung von XML-Daten mittels einer Web-Maske
4. Die (indirekte) Erzeugung von XML-Daten mittels Export aus dem Layout-Programm.

Dabei können die nachfolgenden Kriterien entscheidend für eine Lösung sein:

1. Benutzerbedürfnisse:

Die Gebrauchstauglichkeit (Usability) einer Software setzt sich aus der Effektivität und Effizienz der Anwendungsweise und der Zufriedenheit des Anwenders zusammen und beruht hauptsächlich auf den Bedürfnissen des Benutzers.⁶⁴ Auch wenn die verschiedenen Benutzer divergierende

⁶⁴ Die Definition der Gebrauchstauglichkeit ist in DIN 55350-11, 1995-08, Nr. 4 geregelt und wird hier auf das Gut ‚Software‘ angewandt: „Nach DIN 55350-11, 1995-08, Nr. 4 ist unter Gebrauchstauglichkeit die Eignung eines Gutes zu verstehen im Hinblick auf seinen bestimmungsgemäßen Verwendungszweck; diese Eignung beruht auf objektiv und nicht objektiv feststellbaren Gebrauchseigenschaften. Die Beurteilung der Gebrauchstauglichkeit leitet sich aus individuellen Bedürfnissen ab.“ Zitat aus Quality-Datenbank (2006)

Verhaltensweisen und Arbeitsstile offenbaren, soll dennoch gewährleistet sein, dass sich jeder Anwender möglichst gut in seiner Arbeitsumgebung und in seinem Arbeitsablauf innerhalb des Workflows orientieren kann. Die Erlern- und Bedienbarkeit der Arbeitswerkzeuge soll daher weitgehend intuitiv und so einfach wie möglich gehalten werden.

- a. Vorlage – Dem Autor wird eine Vorlage für die Inhaltserstellung bereitgestellt. Wird die Vorlage auf das Dokument angewendet, kann sich die Ansicht des Dokuments verändern. Daraus folgt eine Einarbeitung des Autors in die veränderte Arbeitsweise. Welche Art von Vorlage ist für den Autor vorgesehen? Verändert sich das Erscheinungsbild der Dokumentenoberfläche durch die Anwendung dieser Vorlage?
- b. Dokumentation – Ist eine Dokumentation standardmäßig vorhanden oder muss sie zusätzlich erstellt werden?

2. Installation und Datenaustausch

- a. Software-Installation – Muss für die XML-Erzeugung ein Zusatzprogramm installiert und ggf. angeschafft werden?
- b. Formelintegration – Wie werden Formeln über die XML-Schnittstelle integriert (in MathML-, Unicode- oder Bild-Kodierung)?
- c. Bildintegration – Werden Bildpfade automatisch in das XML-Dokument eingebunden oder müssen sie manuell eingegeben werden?
- d. Speicherort – Werden die XML-Dokumente automatisch in einem Datenhaltungssystem abgelegt oder müssen sie manuell eingepflegt werden?

Kosten

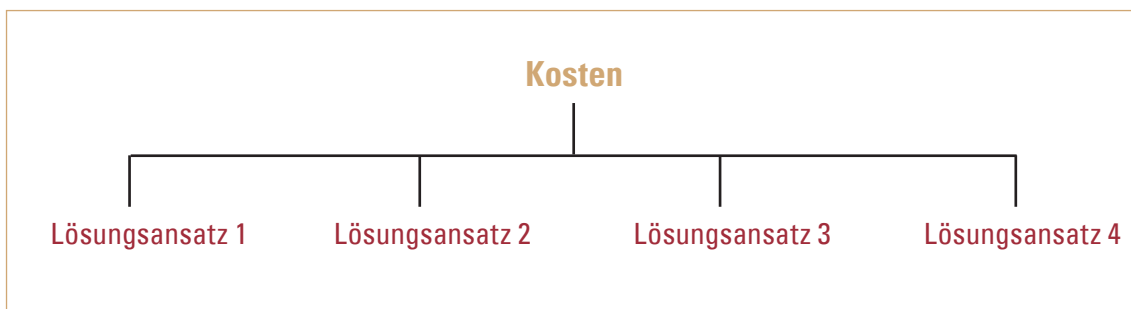


Abb. 18: Lösungsmodul ‚Kosten‘⁶⁵

Kosten können durch die Neuanschaffung notwendiger Software oder Updates sowie durch höheren Personalbedarf oder erhöhten Arbeitsaufwand durch die Implementierung einer solchen Lösung entstehen. Da die Abschnitte der Lösungsansätze jeweils zu einem Lösungsweg zusammengesetzt werden, entstehen unterschiedliche Kosten, die hier im Folgenden berücksichtigt werden; allerdings wird danach unterschieden, ob die Kosten zunächst auf Autoren- und Herausgeberseite anfallen.

⁶⁵ Die vier verschiedenen Lösungsansätze wurden in Abbildung 12 detailliert dargestellt.

1. Kosten auf Autorensseite

a/b. Lizenz/Upgrade – Welche Kosten entstehen bei der Anschaffung von Software? Muss eine Lizenz und gegebenenfalls ein Update der Produktversion erworben werden, um den vollen und notwendigen Funktionsumfang zu erhalten?

2. Kosten auf Herausgeberseite

a/b. Lizenz/Upgrade – Auch beim Herausgeber können Kosten für eine neue Software oder ein notwendiges Update entstehen.

c. Personalbedarf im Workflow – Welche Kapazität an Personal mit welcher Qualifikation ist erforderlich?

d. Implementierungsaufwand – Wie hoch ist der Aufwand, der durch die Realisierung eines solchen Lösungsweges entsteht?

e. Personalbedarf für Implementierung – Wird zusätzliches Personal für die Implementierung erforderlich und welche Qualifikation muss das Personal haben? Genügt der Einsatz einer einfachen Bürokräft oder muss speziell geschultes bzw. fachspezifisches Personal hinzugezogen werden?

Prozessablauf

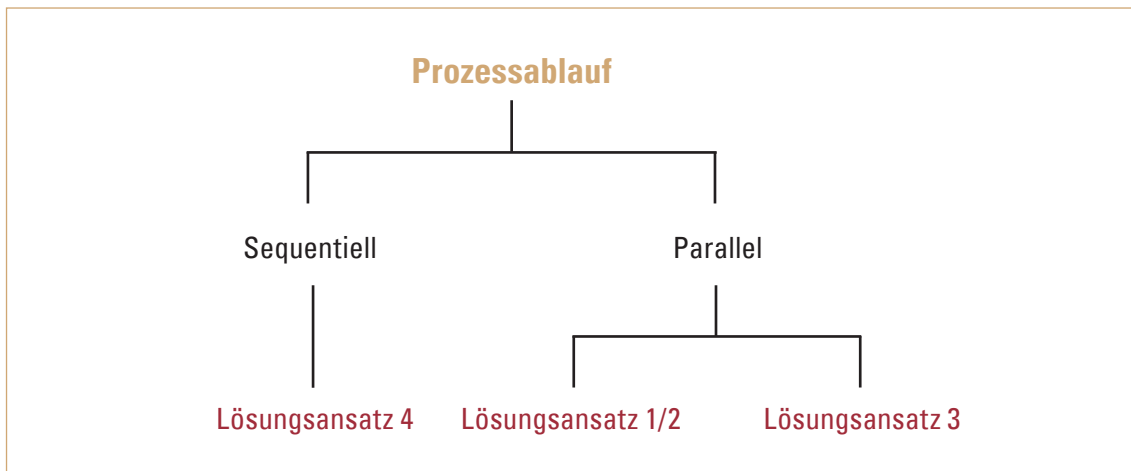


Abb. 19: Lösungsmodul ‚Prozessablauf‘⁶⁶

Der Prozessablauf kann, wie bereits in Kapitel 3.4 deutlich gemacht wurde, einen unterschiedlichen Verlauf nehmen. Über die folgenden Kriterien sowie deren anschließende Bewertung und Gewichtung soll schließlich auch die Entscheidung gefällt werden, ob bestimmte im Prozess vorhandene Datenprobleme gelöst werden. Damit verbunden ist die Frage, nach der geeigneten Reihenfolge der Prozesselemente (parallel oder sequentiell ablaufende Prozessschritte). Die Kriterien ergeben sich aus dem Anforderungsprofil; sie sind zwingend erforderlich, um eine Entscheidung über den geeigneten Produktionsablauf herbeiführen zu können.

⁶⁶ Die vier verschiedenen Lösungsansätze wurden in Abbildung 12 detailliert dargestellt.

1. Datenaustausch

- a. Datenübergabe an Print Workflow – Werden Satzprobleme, wie sie in der Ausgangssituation vorhanden sind, gelöst?
- b. Datenübergabe an Web Workflow – Werden Satzprobleme, wie sie in der Ausgangssituation vorhanden sind, gelöst?
- c. Datenhaltung – Kann ein Datenhaltungssystem für Bilder in den Ablauf integriert werden?

2. Qualitätssicherung

- a. Gestaltungsanspruch – Wird der Gestaltungsanspruch des Print-Produktes erfüllt? Ist es möglich die kreativen Ideen des Layouters in die Gestaltung des Print-Produktes einzubeziehen? Kann eine bestimmte Print-Ausgabe gegenüber einer anderen über veränderte Gestaltungselemente unterschieden werden?
- b. Automatisierung – Ist der Web Workflow automatisiert in den bestehenden Workflow integriert worden?
- c. Freigabezyklus – Sind die Fehlerquellen über einen veränderten Freigabezyklus reduziert worden?

4 Evaluation Teil II – Bewertung und Ergebnis

Der Vorgang der Evaluation besteht neben dem Festlegen von Beurteilungskriterien und Evaluationsobjekten (Evaluation Teil I) insbesondere aus der Durchführung der eigentlichen Bewertung (Evaluation Teil II). Dazu bedarf es eines geeigneten methodischen Vorgehens. Für die hier angestrebte Evaluation der Handlungsoptionen zur Publikation wissenschaftlicher Periodika bietet sich die Nutzwertanalyse aufgrund ihrer einfachen und flexiblen Anwendung an. Sie wird deshalb im Folgenden für die Bewertung herangezogen. Das Resultat einer solchen Analyse besteht in der Zuweisung von so genannten Nutzwerten für die einzelnen Evaluationsobjekte. D.h., ein höherer Nutzwert signalisiert, dass dieses Bewertungsobjekt – unter den getroffenen Prämissen (Kriterien, Gewichtungen etc.) – eine größere Eignung für die beschriebene Problemstellung besitzt. Am Ende der Analyse ergibt sich eine Rangordnung der bewerteten Entscheidungsalternativen. Der beste Lösungsansatz erhält dabei den höchsten Nutzwert.

Kapitel 4, welches den zweiten Teil der Evaluation darstellt, besteht aus der Bewertung und dem Ergebnis. Die Bewertung gliedert sich wiederum in drei Abschnitte. Eine Diskussion der Vor- und Nachteile einzelner Lösungsmodule wird in 4.1 geführt. In Kapitel 4.2 werden den Lösungselementen Nutzwerte zugewiesen, welche anschließend in Kapitel 4.3 gewichtet werden. Das Ergebnis setzt sich aus der Interpretation einzelner Bewertungsergebnisse der Lösungsmodule (4.4) sowie aus der Gesamtinterpretation der Evaluation (4.5) zusammen.

4.1 Beurteilung der einzelnen Lösungsmodule

Eine vergleichende Beurteilung der Evaluationsobjekte – d.h. eine bewertende Gegenüberstellung der möglichen Lösungselemente – setzt voraus, dass die Alternativen inhaltlich normalisiert werden. Dies geschieht, indem sie denselben Bewertungskriterien unterworfen werden. Die bereits im Kriterienkatalog detailliert entwickelten Bewertungsmerkmale werden im Folgenden auf die einzelnen Lösungselemente angewendet. Die Beurteilung der Kriterien erfolgt bei der Inhaltserstellung aus Sicht der Autoren. Die Merkmale der Module der XML-Erzeugung, des Prozessablaufes und der Kosten werden nach quantitativen oder qualitativen Nutzen sowie nach Wichtigkeit und Dringlichkeit des Entscheidungsträgers, folglich aus der Sicht des Herausgebers bewertet.

Nachfolgend werden nun die Vor- und Nachteile der vorgestellten Lösungselemente eines Moduls diskutiert. Im Anschluss an jedes Modul folgt eine Tabelle, in der die Ausprägungen der Bewertungskriterien vergeben werden.

Formelherstellung

Ein wichtiger Vorteil für den Anwender bei der Erstellung von mathematischen Ausdrücken ist die Möglichkeit, die Formel als Ganzes zu bearbeiten (z.B. per Doppelklick auf die Formel). Die

Bearbeitung des Ausdrucks mittels Veränderung einzelner Zeichen stellt demgegenüber die deutlich schlechtere, weil aufwendigere und damit auch langsamere Alternative dar.

Auch die Eingabe über eine übersichtliche Symbolleiste, welche die verfügbaren mathematischen und technischen Zeichen grafisch bereitstellt, bietet dem Nutzer wesentliche Vorteile in der Anwendbarkeit. Eine Zeichentabelle, aus der jedes einzelne Formelzeichen herausgesucht und dem Text mittels ‚Copy&Paste‘ hinzugefügt werden muss, besitzt diese unterstützende Wirkung nicht und hemmt den Anwender in seiner Tätigkeit.

Eine weitere, wertvolle Unterstützungsfunktion stellt die dem Eingabeeditor beigefügte Dokumentation dar. Sie bietet dem Autor schnelle und unabhängige Hilfestellung bei Fragen und Problemen. Der Herausgeber ist dadurch nicht gezwungen, für jeden Publikationsvorgang eine eigenständige Dokumentation erstellen zu müssen. Dies ist von Vorteil, da es die Kosten für den Publikations-Workflow reduziert.

Als besonders schwerwiegender Nachteil bei der Formelerstellung ist hingegen die Notwendigkeit, über spezifisches Zusatzwissen zu verfügen, anzusehen. Dies ist beispielsweise der Fall bei der Eingabe von Unicode mittels Tastatur. Des Weiteren ist es für den Anwender unangenehm, wenn er nicht nur ein Zusatzprogramm zur Formelerstellung installieren, sondern sich dafür auch noch eine Lizenz und gegebenenfalls ein Produkt-Update beschaffen muss.

Word – Formelerstellung	Symbol einfügen	Unicode über Tastatur	MS Fomeleditor	MathType
Anwenderfreundlichkeit				
Eingabe	Zeichentabelle	manuell über die Tastatur	Symbolleiste	Symbolleiste
Bearbeitung	Einzelne Zeichen	Einzelne Zeichen	Formel im Ganzen	Formel im Ganzen
Zusatzinformation	nicht erforderlich	erforderlich	nicht erforderlich	nicht erforderlich
Dokumentation	vorhanden	vorhanden	vorhanden	vorhanden
Software				
Installation (bei Autor)	nicht erforderlich	nicht erforderlich	erforderlich	erforderlich
FormelAusgabe				
Formelkodierung	Unicode	Unicode	GIF/TIFF	MathML/GIF
Kosten auf Autorensseite				
Lizenz	nicht erforderlich	nicht erforderlich	nicht erforderlich	pro Autor erforderlich (ca 76 €)
Upgrade	nicht erforderlich	nicht erforderlich	nicht erforderlich	ca. 40 € pro Autor

Tabelle 2a: Beurteilung der einzelnen Lösungsmodule: Formelerstellung in Word

Web-Maske – Formelerstellung	JavaScript-Anwendung	Java-Applet (WebEQ)
Anwenderfreundlichkeit		
Eingabe	Zeichentabelle	Symbolleiste
Bearbeitung	einzelne Zeichen	Formeln im Ganzen
Zusatzinformation	nicht erforderlich	nicht erforderlich
Dokumentation	nicht vorhanden	nicht vorhanden
Software		
Installation (bei Autor)	nicht erforderlich	nicht erforderlich
FormelAusgabe		
Formelkodierung	Unicode	MathML/GIF/JPEG/PNG
Kosten auf Autoreiseite		
Lizenz	nicht erforderlich	nicht erforderlich
Upgrade	nicht erforderlich	nicht erforderlich

Tabelle 2b: Beurteilung der einzelnen Lösungsmodule: Formelerstellung über eine Web-Maske

Formelkodierung

Ein wichtiger Punkt bei der FormelAusgabe ist die korrekte Darstellung im Endprodukt. Die Unterschiede in der Darstellungsweise bei mathematischen Ausdrücken werden bei der Verwendung von Brüchen und Wurzeln besonders deutlich.

Beispiel einer reduzierten (indirekten) Darstellung: a/b oder $\sqrt{a+b}$

Beispiel einer konventionellen (korrekten) Darstellung: $\frac{a+b}{b}\sqrt{a+b}$

Eine korrekte Darstellung macht den mathematischen Ausdruck übersichtlicher sowie strukturierter und ist für den Leser leichter nachzuvollziehen. Eine indirekte Darstellung hat den Nachteil, dass sie bei der Aufbereitung für das Endprodukt in eine korrekte Darstellungsform konvertiert werden muss. Zu einer ordnungsgemäßen Darstellung von Formeln gehören darüber hinaus die Positionierung auf der Grundlinie des Textes sowie die Skalierbarkeit der Formelzeichen mit der Textgröße. Um das korrekte und geforderte Erscheinungsbild im Endprodukt zu erhalten, muss der mathematische Ausdruck stets mit dem Text auf einer Linie stehen und im Fall einer Veränderung an die Textgröße anpassbar sein. Es ist zudem zu berücksichtigen, dass der Quellcode einer Formel bearbeitet werden muss. Von Vorteil ist es, wenn der Benutzer manuell auf den Quellcode zugreifen und ihn direkt bearbeiten kann. Ist ein zusätzliches Programm notwendig, um den Code zu interpretieren und zu bearbeiten, stellt dies eine unkomfortable und zeitaufwendige Lösung dar, da das entsprechende Programm erst einmal vorhanden und verstanden sein muss.

Nach der Erstellung muss die Formel mit dem Textdokument an die DTP-Vorstufe übergeben werden. Da der Textinhalt in XML ausgezeichnet ist, ist es dienlicher, wenn auch der Formelinhalt in die XML-Auszeichnung integriert und daher in einem Dokument vollständig übergeben werden kann.

Nach dem Datenaustausch kann bei der Produktveredelung eine Konvertierung der Formel erforderlich werden, falls die Formel im eingesetzten Anwendungsprogramm (Layout-Programm oder Webbrowser) nicht erkannt bzw. nicht in der geforderten Weise dargestellt wird.

Formelkodierung	Bildformat	Unicode	MathML	SVG
Darstellung				
Layoutdarstellung	konventionel	reduziert	konventionel	konventionel
Formelposition auf Textgrundlinie im Layoutprogramm	ja	ja	-	ja
Formelposition auf Textgrundlinie im Web	nein	ja	ja	schwierig
Skalierbarkeit der Formelgröße mit Textgröße	nein	ja	ja	ja
Darstellung im Webbrowser	ja	ja	über Viewer	über Viewer
Verarbeitung				
Eingabe	MS Formeleditor, MathType	„Symbol einfügen“, „Unicode via Tastatur“, JavaScript-Anwendung	MathType, WebEQ	-
Manuelle Kode-Bearbeitung	nicht möglich	möglich	nicht möglich	nicht möglich
Übergabe	Pfadangabe in XML	eingebettet in XML	Namensraum (MathML) in XML	Namensraum (SVG) in XML
Konvertierung Print	nein	ja	ja	nein
Konvertierung Web	nein	ja	nein	nein
Import in Layoutprogramm	ja	ja	nein	nein
Export aus Layoutprogramm	ja	ja	nein	nur komplette Seite als SVG, nicht in XML integrierbar
Verwertbarkeit Web	nein	nein	ja	nein

Tabelle 3: Beurteilung der einzelnen Lösungsmodule: Formelkodierung

Bildintegration

Die Vorteile bei der Bildintegration beziehen sich hauptsächlich auf die richtige Bildplatzierung im Textfluss, auf einen möglichst kleinen, d.h. speicherentlastenden Datenumfang bei gegebener Darstellungsqualität sowie auf die Möglichkeit, das ursprüngliche Bildformat im Dokument zu erhalten bzw. wieder herstellen zu können. Die Übergabe der Inhalte zur Weiterverarbeitung ist für einen Autor dann besonders übersichtlich und damit vorteilhaft, wenn er lediglich ein Dokument übergeben muss, d.h. wenn alle von ihm geforderten inhaltlichen Bestandteile in einer Datei in geeigneter Weise integriert werden können. Bei Verknüpfungen von Bilddaten müssen diese separat als eigenständige Dateien transferiert werden; dies kann bei Unachtsamkeit leicht zum Verlust der Daten führen. Im Falle eines Upload auf den Webserver muss der Autor dem Bildmaterial hingegen keine weitere Aufmerksamkeit widmen.

Bildintegration	Word – Einbettung	Word – Verknüpfung	Web-Maske – Upload
Bildplatzierung	möglich	möglich	möglich
Dateigröße	vergrößert sich	unverändert	unverändert
Übergabe	nur Text-Datei	Text- und Bild-Dateien	keine zusätzl. Übergabe
Speicherort	manuell in DB	manuell in DB	automatisch in DB
Ursprungsformat	evtl. verändert (versionsabh.)	unverändert	unverändert

Tabelle 4: Beurteilung der einzelnen Lösungsmodule: Bildintegration

XML-Erzeugung

Die bisherigen Ausführungen verdeutlichen, dass zum einen die Erstellung des Inhalts durch die Autoren sowohl strukturiert und in direkt verarbeitbarer Form erfolgen muss und zum anderen die Autoren bei der Erzeugung des Inhalts unterstützt und geführt werden sollen. Dies setzt voraus, dass den Autoren geeignete Vorlagen bereitgestellt werden, die diese zwingend nutzen müssen. Dabei kann es sich um eine Dokumentvorlage, um eine Formatvorlage oder um ein exemplarisches XML-Dokument, welches das entsprechende XML-Schema enthält, handeln. Auch eine Web-Maske mit diesbezüglichen Eingabefeldern, über welche die Inhaltselemente direkt strukturiert werden, kann als Vorlage dienen. Für den Autor ist es dabei wichtig, wie die Benutzeroberfläche gestaltet ist.

Die Verwendungsweise von Formatvorlagen dürfte dem geübten Nutzer von Textverarbeitungsprogrammen hinreichend bekannt sein; ein XML-Dokument oder eine Web-Maske sind ihm hingegen erheblich weniger vertraut. Sie weisen für ihn ein neues Erscheinungsbild auf und bedingen dadurch eine zusätzliche Einarbeitungs- und Orientierungszeit. Des Weiteren kann – wie bei der Formelerstellung – eine Dokumentation bei der XML-Erzeugung hilfreich für den Anwender sein. Für den Herausgeber ist es wiederum von Vorteil, wenn eine solche Dokumentation in Form einer Hilfefunktion standardmäßig in die Anwendung integriert ist und somit nicht eigens erstellt werden muss. Nachteilig, sowohl für den Autor als auch den Herausgeber, ist es, wenn ein Zusatzprogramm für die XML-Erzeugung separat angeschafft und installiert werden muss. Dies verursacht zusätzliche Kosten und zeitlichen Aufwand.

Ein besonders wichtiges Merkmal bei der Erzeugung von Inhalten in XML-Form betrifft den Aspekt, wie neben den Textelementen grafische Bestandteile des Inhalts und mathematische Ausdrücke in der XML-Auszeichnung berücksichtigt werden. Bei der Formelintegration muss die Bewertung der Formelaußgabemöglichkeiten berücksichtigt werden, um die Effizienz der jeweiligen Kodierungsart angeben zu können. Bilder werden in XML über ihre Pfadangaben integriert. Dabei ist es zeitsparend, wenn dieses automatisch funktioniert und die Pfade nicht manuell eingegeben und platziert werden müssen. Nachdem die XML-Daten erzeugt und übergeben wurden, müssen sie gespeichert und verwaltet werden. Eine direkte und automatische Speicherung in einem Datenhaltungssystem erspart Aufwand und Zeit (Effizienz) und sorgt für Übersicht und Strukturiertheit (Effektivität).

XML-Erzeugung	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign
Benutzerbedürfnisse				
Vorlage	XML-Dokument	Formatvorlage	Eingabefelder	Formatvorlage
Dokumentation	Word Hilfe	Word Hilfe	nicht vorhanden	Adobe InDesign CS Hilfe
Technologie				
Software-Installation	nicht erforderlich	erforderlich (auf Herausgeberseite)	nicht erforderlich	nicht erforderlich
Formelintegration	GIF	GIF/Unicode	MathML/GIF/Unicode	GIF/Unicode
Bildintegration	automatisch	automatisch	automatisch	automatisch
Speicherort	manuell in DB	manuell in DB	automatisch in DB	manuell in DB

Tabelle 5: Beurteilung der einzelnen Lösungsmodule: XML-Erzeugung

Kosten

Häufig werden höhere Kosten unter der Voraussetzung in Kauf genommen, dass eine Leistungssteigerung erzielt wird. Die entscheidende Relation ist deshalb jene zwischen den verursachten Kosten und der erwirkten Leistung. Für den Fall, dass eine Publikation von der Beitragseinreichung durch einen offenen und unabhängigen Autorenkreis abhängt, der keine oder eine nur geringe finanzielle Aufwandsentschädigung für diese Beiträge erhält, gilt es, möglichst jegliche Kosten für die Autoren zu vermeiden. D.h., dem Autor sollen keine Kosten zugemutet werden, es sei denn er wendet sie aus eigenem Interesse auf.

Auf der Herausgeberseite sieht die Sachlage anders aus. Die Bereitschaft, Kosten für eine effizientere Produktion aufzuwenden, ist vorauszusetzen. Bieten sich jedoch mehrere Lösungswege an, von denen zu vermuten ist, dass sie alle eine Verbesserung der Produktion ermöglichen, so gilt die Präferenz der Entscheidungsträger der Kostenbegrenzung: sie werden in der Regel die kostengünstigste Variante auswählen.

Kosten	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign
Kosten auf Autorensseite				
Lizenz	nein	nein	nein	nein
Upgrade	evtl. erforderlich bei einzeltem Autor (399 €)	nein	nein	nein
Kosten auf Herausgeberseite				
Lizenz	nein	Konverter (0-330,60 €)	WebEQ (450 €)	nein
Upgrade	nein	evtl.	69 €	nein
Personalbedarf im WF	erforderlich (XML-Kontrolle)	erforderlich (XML-Generierung)	nicht erforderlich	nicht erforderlich
Implementierungsaufwand	mittel	mittel	hoch	gering
Personal für Implementierung	nicht erforderlich	nicht erforderlich	+1 (Webentwickler)	nicht erforderlich

Tabelle 6: Beurteilung der einzelnen Lösungsmodule: Kosten

Prozessablauf

Die Vorteile einer gezielten Workflow-Gestaltung zeigen sich insbesondere in einem verbesserten Datenaustausch sowie einer gesicherten Produktqualität. Durch das Datenaustauschformat XML werden die inhaltlichen Bestandteile der Daten von ihrer Darstellungsweise getrennt. Textformatierungen und andere Gestaltungsanweisungen, die Satzprobleme verursachen können, werden folglich nicht mit dem Inhalt übergeben. Auf diese Weise lassen sich die angesprochenen Satzprobleme vermeiden.

Die Qualität wird durch eine sinnvolle Verteilung der Korrekturschritte des Freigabezyklus gewährleistet. Ab dem Punkt, an dem eine Kontrolle abgeschlossen werden kann, können auch keine Fehler mehr entstehen; das bedeutet je früher dieser Zeitpunkt eintritt, desto höher die Produktqualität. Der inhaltliche Freigabeprozess ist beispielsweise mit der Schaffung eines medienneutralen Datenbestandes abgeschlossen, die Daten müssen folglich nicht mehr auf Inhalte kontrolliert werden. In einem parallel verlaufenden Prozess ist der Freigabezyklus folglich optimal angelegt. Durch einen automatisierten Web Workflow wird die Qualität insofern verbessert, als dass sich dort keine Fehler mehr bilden können. Auf der anderen Seite muss beachtet werden, dass sich gerade das in dieser Arbeit betrachtete Druckprodukt durch eine anspruchsvolle Gestaltung auszeichnet. Die Erfüllung dieser Anforderung wirkt sich nicht nur positiv auf die Qualität aus, sondern sie ist notwendige Voraussetzung, um im Publikationsmarkt bestehen zu können.

Prozessablauf	Parallel (Word)	Parallel (Web-Maske)	Sequentiell (InDesign)
Datenübergabe			
Datenübergabe an Print WF	Satzprobleme durch XML gelöst	Satzprobleme durch XML gelöst	Satzprobleme über Vorlage und durch Import vermindert
Datenübergabe an Web WF	Satzprobleme durch XML gelöst	Satzprobleme durch XML gelöst	Satzprobleme durch XML gelöst
Datenhaltung	Bilddatenbank möglich	Bilddatenbank möglich	Bilddatenbank möglich
Qualitätssicherung			
Gestaltungsanspruch Print	gegeben	gegeben	gegeben
Automatischer Web WF	teil-automatisiert	voll-automatisiert	teil-automatisiert
Freigabezyklus	optimiert	optimiert	verbessert

Tabelle 7: Beurteilung der einzelnen Lösungsmodule: Prozessablauf

4.2 Überführung der Merkmalsausprägungen in Nutzwerte

Die einzelnen Ausprägungen in den verschiedenen Bewertungsmerkmalen werden mittels Skalierung in entsprechende Nutzwerte überführt. Zur Wahl stehen nominale und ordinale Skalen.⁶⁷ Die Nominalskala lässt eine Entweder-Oder-Kategorie zu und wird auf die Fragen des Kriterienkataloges angewandt, die ‚Ja‘ oder ‚Nein‘ als Antwort haben. Die Rangskala lässt einen Vergleich von verschiedenen Merkmalen zu, ohne den Abstand zwischen den Rängen kennen zu müssen. So können die Kriterienwerte in eine Rangfolge eingeteilt und darüber miteinander verglichen werden.

Die Nutzwerte, ob nominal oder ordinal gemessen, müssen in einer festen Relation zueinander vergeben werden, damit sie miteinander vergleichbar sind und zu einer Gesamtpunktzahl summiert werden können. Die Verteilung wurde in dieser Analyse so gewählt, dass jedem Merkmal minimal 0 und maximal 3 Punkte zugeteilt werden können. Eine positive Ausprägung auf der Nominalskala erhält 3 Punkte, für eine negative Ausprägung werden 0 Punkte vergeben. In der Ordinalskala wird die Merkmalsausprägung, welche das Kriterium am besten erfüllt, mit der maximal zu vergebenden Punktzahl von 3 bemessen, bei Nichterfüllung des Kriteriums gibt es keinen Punkt. Bei vier verschiedenen Kategorien werden 0, 1, 2 oder 3 Punkte verteilt, bei drei verschiedenen Ausprägungen werden 0, 1.5 oder 3 Punkte zugewiesen. Entsprechend dieser Festlegung werden in den folgenden Tabellen die Merkmalsausprägungen in Nutzwerte umgewandelt.

⁶⁷ Vgl. Heinrich/Lehner (2005), S. 382 f

4. Evaluation Teil II – Bewertung und Ergebnis

Formelkodierung	max.	Bildformat	Unicode	MathML	SVG
Darstellung					
Layoutdarstellung (Brüche/Wurzeln/Klammern)	3	3	0	3	3
Formelposition auf Textgrundlinie im Layoutprogramm	3	3	3	0	3
Formelposition auf Textgrundlinie im Web	3	0	3	3	1,5
Skalierbarkeit der Formelgröße mit Textgröße	3	0	3	3	3
Darstellung im Webbrowser	3	3	3	1,5	1,5
Verarbeitung					
Eingabe	-	-	-	-	-
Manuelle Kode-Bearbeitung	3	0	3	0	0
Übergabe	3	1,5	3	3	3
Konvertierung Print	3	3	0	0	3
Konvertierung Web	3	3	0	3	3
Import in Layoutprogramm (Verwertbarkeit Print)	3	3	3	0	0
Export aus Layoutprogramm	3	3	3	0	0
Verwertbarkeit Web	3	0	0	3	0
Zwischensumme	36	22,5	24	19,5	21

Tabelle 8: Überführung der Merkmalsausprägungen in Nutzwerte: Formelkodierung

Formelerstellung	max.	Symbol einfügen	Unicode via Tastatur	MS Formeleditor	MathType	JavaScript	WebEQ
Anwenderfreundlichkeit							
Eingabe	3	2	1	3	3	2	3
Bearbeitung	3	0	0	3	3	0	3
Zusatzinformation	3	3	0	3	3	3	3
Dokumentation	3	3	3	3	3	0	0
Software							
Installation (bei Autor)	3	3	3	0	0	3	3
Formelausgabe							
Formelkodierung (Übertrag)	36	24	24	22,5	19,5/22,5	24	19,5/22,5
Kosten							
Lizenz	3	3	3	3	0	3	3
Upgrade	3	3	3	3	1	3	3
Zwischensumme	57	41	37	40,5	32,5/35,5	38	37,5/40,5

Tabelle 9: Überführung der Merkmalsausprägungen in Nutzwerte: Formelerstellung

4. Evaluation Teil II – Bewertung und Ergebnis

Bildintegration	max.	Word – einbetten	Word – verknüpfen	Web-Maske – Upload
Bildplatzierung	3	3	3	3
Dateigröße	3	0	3	3
Übergabe	3	1,5	0	3
Speicherort	3	0	0	3
Ursprungsformat	3	1,5	3	3
Zwischensumme	15	6	9	15

Tabelle 10: Überführung der Merkmalsausprägungen in Nutzwerte: Bildintegration

XML-Erzeugung	max.	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign
Benutzerbedürfnisse					
Vorlage	3	0	3	0	3
Dokumentation	3	3	3	0	3
Technologie					
Software-Installation	3	3	0	3	3
Formelintegration	3	3	3	3	3
Bildintegration	3	3	3	3	3
Speicherort	3	0	0	3	0
Zwischensumme	18	12	12	12	15

Tabelle 11: Überführung der Merkmalsausprägungen in Nutzwerte: XML-Erzeugung

Kosten	max.	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign
Kosten auf Autorensseite					
Lizenz	3	3	3	3	3
Upgrade	3	0	3	3	3
Kosten auf Herausgeberseite					
Lizenz	3	3	1,5	0	3
Upgrade	3	3	1,5	0	3
Personalbedarf im WF	3	0	0	3	3
Implementierungsaufwand	3	1,5	1,5	0	3
Personal für Implementierung	3	3	3	0	3
Zwischensumme	21	13,5	13,5	9	21

Tabelle 12: Überführung der Merkmalsausprägungen in Nutzwerte: Kosten

Prozessablauf	max.	Parallel (Word)	Parallel (Web-Maske)	Sequentiell (InDesign)
Datenübergabe				
Datenübergabe an Print WF	3	3	3	1,5
Datenübergabe an Web WF	3	3	3	3
Datenhaltung	3	3	3	3
Qualitätssicherung				
Gestaltungsanspruch Print	3	3	3	3
Automatischer Web WF	3	1,5	3	1,5
Freigabezyklus	3	3	3	1,5
Zwischensumme	18	16,5	18	13,5

Tabelle 13: Überführung der Merkmalsausprägungen in Nutzwerte: Prozessablauf

4.3 Gewichtung der Bewertungskriterien

Die Bewertungskriterien haben für die Entscheidungsträger im Allgemeinen eine unterschiedlich große Bedeutung, daher werden die Merkmale gewichtet, d.h. es wird eine Präferenzordnung der Kriterien hergestellt und in Koeffizienten abgebildet.⁶⁸ Die Bewertungsmerkmale werden dazu in drei Bedeutungsklassen eingeteilt. Für Kriterien der höchsten Präferenz wird der ermittelte Nutzwert mit dem Koeffizienten drei gewichtet. Die Nutzwerte von Merkmalen mittlerer Bedeutung werden mit dem Faktor zwei multipliziert. Die Nutzwertausprägung von Merkmalen mit geringem Gewicht für die Beurteilung der Güte einer Lösungsalternative behält ihren Nutzwert (Koeffizient 1).

Höchste Präferenzgruppe (Koeffizient 3)

Die Erfüllung eines Kriteriums dieser Präferenzkategorie ist besonders wichtig für die Weiterverarbeitung im Workflow. Zu dieser Gruppe zählen u.a. Bewertungsmerkmale wie der Anspruch auf individuelle Gestaltung und die Kosten für den Implementierungsaufwand.

Mittlere Präferenzgruppe (Koeffizient 2)

Dieser Kategorie gehören die Kriterien an, die die Anwender – und hier in erster Linie die Autoren – betreffen. Neben der Anwenderfreundlichkeit zählen auch die Software-Kosten dazu, die nicht nur für den Autor, sondern auch für den Herausgeber entstehen können.

Niedrige Präferenzgruppe (Koeffizient 1)

In diese Kategorie fallen Kriterien, welche vor allem mit Problemen wie Dateigrößen und zusätzlichen Programminstallationen zu tun haben. Beim Datenaustausch bspw. stellt jedoch die Dateigröße, bedingt durch die schnellen Internetübertragungsleitungen, kein Hindernis mehr dar. Der Aufwand einer kleinen Installation zur Darstellungsunterstützung im Browser oder zur

⁶⁸ Vgl. Heinrich/Lehner (2005), S. 382 f

Formeleingabe ist mittlerweile ebenfalls zu vernachlässigen. Die meisten Installationsdateien lassen sich bequem über das Internet herunterladen und per Mausklick automatisch entpacken und installieren. Gemeinsam haben die Kriterien dieser Gruppe also, dass sie jeweils nicht oberste Priorität aufweisen.

Die Gewichtung der einzelnen Kriterien und Lösungswege ist hier nochmals anschaulich in Form von Tabellen vorzufinden.

Formelkodierung	Faktor	Bildformat	Unicode	MathML	SVG
Darstellung					
Layoutdarstellung (Brüche/Wurzeln/Klammern)	3	9	0	9	9
Formelposition auf Textgrundlinie im Layoutprogramm	2	6	6	0	6
Formelposition auf Textgrundlinie im Web	2	0	6	6	3
Skalierbarkeit der Formelgröße mit Textgröße	2	0	6	6	6
Darstellung im Webbrowser	1	3	3	1,5	1,5
Verarbeitung					
Eingabe	-	-	-	-	-
Manuelle Kode-Bearbeitung	1	0	3	0	0
Übergabe	3	4,5	9	9	9
Konvertierung Print	3	9	0	0	9
Konvertierung Web	3	9	0	9	9
Import in Layoutprogramm (Verwertbarkeit Print)	3	9	9	0	0
Export aus Layoutprogramm	3	9	9	0	0
Verwertbarkeit Web	3	0	0	9	0
Ergebnis (max. 87)		58,5	51	49,5	52,5

Tabelle 14: Gewichtung der Bewertungskriterien: Formelkodierung

4. Evaluation Teil II – Bewertung und Ergebnis

Formelerstellung	Faktor	Symbol einfügen	Unicode via Tastatur	MS Formel-editor	MathType	JavaScript	WebEQ
Anwenderfreundlichkeit							
Eingabe	2	4	2	6	6	4	6
Bearbeitung	2	0	0	6	6	0	6
Zusatzinformation	2	6	0	6	6	6	6
Dokumentation	2	6	6	6	6	0	0
Software							
Installation (bei Autor)	1	3	3	0	0	3	3
Formelkodierung							
Formelkodierung (Übertrag)		51	51	58,5	49,5/58,5	51	49,5/58,5
Kosten							
Lizenz	2	6	6	6	0	6	6
Upgrade	2	6	6	6	2	6	6
Ergebnis (max. 126)		82	74	94,5	75,5/84,5	76	82,5/91,5

Tabelle 15: Gewichtung der Bewertungskriterien: Formelerstellung

Bildintegration	Faktor	Word – einbetten	Word – verknüpfen	Web-Maske – Upload
Bildplatzierung	3	9	9	9
Dateigröße	1	0	3	3
Übergabe	2	3	0	6
Speicherort	2	0	0	6
Ursprungsformat	3	4,5	9	9
Ergebnis (max. 33)		16,5	21	33

Tabelle 16: Gewichtung der Bewertungskriterien: Bildintegration

XML-Erzeugung	Faktor	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign
Benutzerbedürfnisse					
Vorlage	2	0	6	0	6
Dokumentation	2	6	6	0	6
Technologie					
Software-Installation	1	3	0	3	3
Formelintegration	3	9	9	9	9
Bildintegration	3	9	9	9	9
Speicherort	2	0	0	6	0
Ergebnis (max. 39)		27	30	27	33

Tabelle 17: Gewichtung der Bewertungskriterien: XML-Erzeugung

Kosten	Faktor	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign
Kosten auf Autorensseite					
Lizenz	2	6	6	6	6
Upgrade	2	0	6	6	6
Kosten auf Herausgeberseite					
Lizenz	2	6	3	0	6
Upgrade	2	6	3	0	6
Personalbedarf im WF	2	0	0	6	6
Implementierungsaufwand	3	4,5	4,5	0	9
Personal für Implementierung	3	9	9	0	9
Ergebnis (max. 48)		31,5	31,5	18	48

Tabelle 18: Gewichtung der Bewertungskriterien: Kosten

Prozessablauf	Faktor	Parallel (Word)	Parallel (Web-Maske)	Sequentiell (InDesign)
Datenübergabe				
Datenübergabe an Print WF	3	9	9	4,5
Datenübergabe an Web WF	3	9	9	9
Datenhaltung	3	9	9	9
Qualitätssicherung				
Gestaltungsanspruch Print	3	9	9	9
Automatischer Web WF	3	4,5	9	4,5
Freigabezyklus	3	9	9	4,5
Ergebnis (max. 54)		49,5	54	40,5

Tabelle 19: Gewichtung der Bewertungskriterien: Prozessablauf

4.4 Interpretation der Bewertungsergebnisse

Eine funktionelle Beurteilung einzelner Vorgehensoptionen wurde in Kapitel 4.1 ausgearbeitet. Nachdem die Evaluation methodisch festgelegt wurde und die Bewertungsmerkmale in Nutzwerte überführt und nach ihrer Präferenz gewichtet wurden, ist es nun notwendig die einzelnen Ergebnisse zu interpretieren.

An dieser Stelle sei noch einmal darauf hingewiesen, dass die Frage, ob der Publikationsprozess parallel oder sequentiell erfolgt, weiterhin offen bleibt. Diese Entscheidung kann erst dann getroffen werden, wenn die Ergebnisse aus der Bewertung der einzelnen Module interpretiert wurden und der Workflow modular aus den bestmöglichen Lösungselementen zusammengesetzt wurde. Aus diesem Grund wird in der Interpretation nicht nur auf die Elemente mit der absolut höchsten Punktzahl eingegangen, sondern auch auf die Elemente, welche jeweils die Höchstpunktzahl bei alternativer Vorgehensoption erreichen.

Formelerstellung

Ein wichtiger Bestandteil der Formelerstellung ist die Kodierung des mathematischen Ausdrucks. Daher basiert das Bewertungskriterium ‚Formelkodierung‘ auf den Resultaten der separaten Beurteilung verschiedener Formate (MathML, Unicode, SVG und Rastergrafiken) von Formeln. Das Ergebnis dieser Bewertung zeigt, dass mathematische Ausdrücke, die im Bildformat ausgegeben werden, beim derzeitigen technologischen Stand am Besten zu handhaben sind, obwohl sie technologisch betrachtet keine besonders fortschrittliche Lösung darstellen. Unter anderem beruht dies darauf, dass die richtige Darstellung von Bruchzeichen, Wurzelausdrücken oder Klammern in mathematischen Komplexen für Print- und Web-Produkte ausschlaggebend ist. Unicode-basierte Formeln sind zwar vergleichsweise einfach zu erstellen, zu verarbeiten und zu übergeben. Die reduzierte Darstellungsform und die daraus resultierende Notwendigkeit der aufwendigen Bearbeitung und Konvertierung, um zu der geforderten konventionellen Darstellungsweise zu gelangen, stellt allerdings einen schwerwiegenden Nachteil dar.

Die Ausgabe in MathML ist durchaus eine attraktive Lösung und erfüllt den Anspruch an eine korrekte Darstellungsweise. Da dieses Format jedoch von den eingesetzten Anwendungsprogrammen – wie dem Layout-Programm und dem Webbrowser – noch unzureichend unterstützt wird, kann sie für eine Formelausgabe zum gegenwärtigen Zeitpunkt nicht empfohlen werden.

Für die Bewertung der Formelerstellung hat sich die Formeleingabe in Word über den MS Formel-Editor mit 94,5 von 126 erreichbaren Punkten durchgesetzt. Formeln werden dabei in Bildformaten ausgegeben; zusätzliche Kosten entstehen nicht, lediglich eine kleine Zusatzinstallation des Office-Paketes wird seitens des Autors erforderlich.

Der Einsatz einer Web-Maske mit dem Java-Applet ‚WebEQ Input Control‘, das zur Ausgabe in MathML auch die Ausgabe in Bildformate ermöglicht, erreicht 91,5 Punkte. Mathematische Ausdrücke lassen sich folglich auch sehr gut über einen Webbrowser eingeben und verarbeiten. Dabei ist jedoch die Formelausgabe als Bildformat entscheidend, die in der Weiterverarbeitung ihren Zweck erfüllt. Die Ausgabe in MathML steht im Gegensatz dazu; diese Formelauszeichnung kann im Layout-Programm derzeit nicht interpretiert und verarbeitet werden. Unter Betrachtung dieser Formelkodierung erreichte die Eingabeanwendung lediglich 82,5 Punkte und sollte daher nicht in einer Empfehlung berücksichtigt werden.

Bildintegration

Das Ergebnis der Bewertung der Bildintegration ergab, dass der Upload von Bildern über eine Web-Maske mit voller Punktzahl (33 Punkte) eindeutige Vorteile gegenüber den Möglichkeiten von Word bietet. Somit wird hier empfohlen, Bilder über den Einsatz einer Web-Anwendung direkt in die Datenbank zu laden. Das Bildmaterial kann dadurch weder verändert werden noch muss es separat zu der Textdatei übergeben werden. Im Fall eines Lösungsweges, der auf der

Inhaltserstellung über Word erfolgt und keine Web-Anwendung vorsieht, sollten Bilder verknüpft und separat überliefert werden. Die Verknüpfung von Bildern mit Word wurde mit 21 Punkten bewertet, die Einbettung erreichte lediglich 16,5 Punkte.

XML-Erzeugung

Die höchste Punktzahl in der Bewertung der XML-Erzeugung erreichte die Möglichkeit des sequentiellen Lösungsweges über das Layout-Programm ‚InDesign CS‘ mit 33 von 39 möglichen Punkten. Diese Vorgehensweise der XML-Erzeugung erfolgt im Anschluss an die Inhaltserstellung über eine Formatvorlage in Word und der Aufbereitung sowie der Fertigstellung der Daten zum Print-Produkt. Mit geringem Aufwand lässt sich eine Mustervorlage im Layout-Programm erstellen, die die notwendigen XML-Tags beinhaltet. Die Tags werden den entsprechenden Stilvorlagen des Layout-Dokuments zugeordnet, so dass die Vorlage für einen wiederkehrenden Prozess erneut verwendet werden kann. Bei der Wahl eines parallelen Workflow hat die Erzeugung von XML-Daten aus Word über den zusätzlichen Konverter 30 Punkte erreicht und liegt somit vor der Option der XML-Generierung über die Produktversion ‚Professional 2003‘ und dem Einsatz einer Web-Maske, die jeweils 27 Punkte erzielten. Aufgrund der Benutzerbedürfnisse wird die Aufbereitung von XML-Daten erstaunlicherweise aus dem Layout-Programm besser als die Aufbereitung aus Word oder der Web-Anwendung bewertet. Dies zeigt jedoch, wie schon in Kapitel 3.4 angenommen wurde, dass die Planung und die Erzeugung eines XML-basierten Datenbestandes zu aufwendig bzw. zu benutzerunfreundlich für den mitwirkenden Autorenkreis sind.

Prozessablauf

Die Bewertungsergebnisse der Ablaufreihenfolge des Publikationsprozesses zeigen, dass ein paralleler Workflow dem sequentiellen gegenüber zu bevorzugen ist. Dabei kann zwischen dem Einsatz einer Web-Anwendung und der Inhaltserstellung mit Word unterschieden werden. Der parallele Ablauf über eine Web-Maske liegt mit 54 von 56 möglichen Punkten eindeutig vor der parallelen Vorgehensweise über Word, die nur 49,5 Punkte erreichte. Der sequentielle Ablauf schnitt mit 40,5 Punkte äußerst schlecht ab. Der parallele Prozessablauf über eine Web-Anwendung wird daher empfohlen.

Kosten

Der sequentiellen Lösungsweg hat bei der Beurteilung anfallender Kosten die volle Punktzahl von 48 erreicht; weder bei der Formelerstellung mit dem MS Formeleditor noch bei der XML-Erzeugung, die über die integrierte Schnittstelle von ‚InDesign CS‘ erfolgt, fallen zusätzliche Kosten an. Da das Layout-Programm im Grunde genommen hierzu bereits vorhanden sein muss, entstehen auch keine weiteren Kosten für dessen Anschaffung. Bei einem parallelen Produktions-Workflow entstehen zusätzliche Kosten. Die geringsten Zusatzkosten ergeben sich bei der Inhaltserstellung mit Word und einem Zusatzprogramm zur Konvertierung der Inhalte in XML-Daten. Wesentlich hierbei ist, dass für die Autoren keine Kosten entstehen. Ähnlich stellt

sich der Fall bei der Lösung über ‚Word Professional 2003‘ dar, bei der sich die Autoren jedoch eventuell ein vergleichsweise teures Upgrade der Office-Version zulegen müssen. Der Aufwand zur Implementierung einer Web-Anwendung ist diesbezüglich am teuersten: der Einsatz eines Web-Entwicklers verursacht die mit Abstand höchsten Kosten.

4.5 Ergebnis der Evaluation

Die Bewertungsergebnisse wurden in Kapitel 4.4 interpretiert und zusammengefasst. Um aber den bestmöglichen Lösungsweg zu identifizieren, muss überprüft werden, welche Lösungselemente sich effizient zu einem kompletten Lösungsweg kombinieren lassen. Teilweise wurde dies durch die vorangegangene Interpretation schon angesprochen. In Abbildung 20 werden noch einmal alle Wege grafisch hervorgehoben. Aus den ursprünglich vier verschiedenen Vorgehensweisen lassen sich noch drei sinnvolle Kombinationen erstellen. Die vierte Zusammensetzung erübrigt sich, da sie sich vom ersten Lösungsweg nur dadurch unterscheidet, dass das XML-Dokument nicht über Word und einem Konvertierungsprogramm, sondern über Word Professional 2003 erzeugt wird. Die Evaluation hat aber gezeigt, dass die erste Möglichkeit der zweiten überlegen ist; daher wird die Anzahl der möglichen Lösungswege auf drei reduziert:

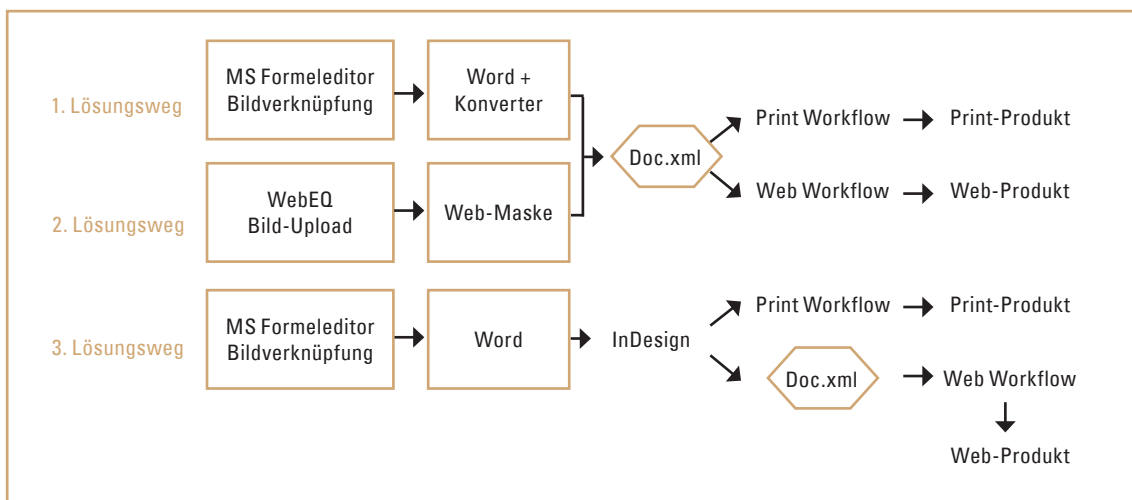


Abb. 20: Die drei Lösungswege, kombiniert aus den jeweils bestmöglichen Elementen der entsprechenden Module

Für jedes Lösungsmodul wird anhand der in Kapitel 4.4 dargestellten Ergebnisse begutachtet, welches Element unter Voraussetzung des jeweiligen Lösungsweges die höchste Punktzahl erreicht. Über die Summe der zusammengesetzten Punktzahlen soll der Lösungsweg mit der höchsten Punktzahl das Ergebnis der Evaluation bilden und damit gleichzeitig die wesentliche Grundlage und Bestandteil für die Ausarbeitung einer geeigneten Konzeption zur Publikation wissenschaftlicher Texte; entsprechende Überlegungen sind Gegenstand des anschließenden Kapitel 5. Eine Übersicht der Punkteverteilung bietet Tabelle 20:

Punkteverteilung						
Einzelne Lösungswege						
	Bild	Unicode	MathML	SVG		
Formelkodierung	58,5	51	49,5	52,5		
	Symbol einfügen	Unicode via Tastatur	MS Formel-editor	MathType (GIF)	JavaSkript	WebEQ (GIF)
Formel-Eingabe	82	74	94,5	84,5	76	91,5
	Word – Einbetten	Word – Verknüpfen	Web-Maske - Upload			
Bildintegration	16,5	21	33			
	Word Prof. 2003	Word + Konverter	Web-Maske	InDesign CS		
XML-Erzeugung	27	30	27	33		
Kosten	31,5	31,5	18	48		
Endverteilung						
Kompletter Lösungsweg	Parallel über Word (Lösung 1)	Parallel über Web-Maske (Lösung 2)	Sequentiell über Word (Lösung 3)			
Inhaltserstellung	115,5	124,5	115,5			
XML-Erzeugung	30	27	33			
Kosten	31,5	18	48			
Prozessablauf	49,5	54	40,5			
Endergebnis (max. 300)	226,5	223,5	237			
Angaben in %	75,5 %	74,5 %	79 %			

Tabelle 20: Punkteverteilung und Endergebnis

Aus den Ergebnissen der einzelnen Lösungsmodule wird deutlich, dass der Einsatz einer Web-Anwendung sowohl für die Inhaltserstellung als auch für die Reihenfolge der Weiterverarbeitung (Prozessablauf) als beste Lösung zu empfehlen ist. In Bezug auf die XML-Erzeugung und die hierdurch entstehenden Zusatzkosten erweist sich der sequentielle Lösungsweg (in Abbildung 20 als 3. Lösungsweg dargestellt) als deutlich vorteilhafter.

Bei der Kombination einzelner Punktergebnisse zu einem kompletten Lösungsweg zeigt sich eine etwas unglückliche Punkteverteilung, da die so ermittelten aggregierten Nutzwerte sehr dicht beieinander liegen und eine eindeutige Lösungsempfehlung dadurch erschweren. Dabei erreicht der sequentielle Verarbeitungsablauf (s. Abbildung 20, 3. Lösungsweg) 237 Punkte, der parallele Weg über die Inhaltserstellung mit Word (s. Abbildung 20, 1. Lösungsweg) 226,5 Punkte und die zweite parallele Lösungsmöglichkeit über den Einsatz einer Web-Maske (s. Abbildung 20, 2. Lösungsweg) 223,5 Punkte von jeweils 300 möglichen Nutzwertpunkten.

Die unterschiedlichen Lösungswege weisen unterschiedliche Vor- und Nachteile auf. Eine bestmögliche Lösung, die von der Punktzahl innerhalb des Toleranzintervalls zwischen 90 und 100 Prozent der Maximalpunktzahl liegen sollte, kann insofern nicht ermittelt werden. Auf Grund

der nahe beieinander liegenden Ergebnisse ist eine abschließende Handlungsempfehlung vom spezifischen Anwendungsfall, also vom Beispiel ‚inSiDE‘ abhängig zu machen. Prinzipiell sollte bei einer so engen Punktevergabe die Entscheidung auf eine Lösungsmöglichkeit fallen, die Medienneutralität gewährleisten kann und folglich parallel verlaufen muss. Daher ist es sehr überraschend, dass sich der parallele Weg nicht von einem sequentiellen Ablauf abheben kann. Die Beurteilung hat eindeutig gezeigt, dass auch nach dem sequentiellen Lösungsansatz der bestehende Print-Publikations-Workflow verbessert und der Web Workflow automatisiert umgesetzt werden kann.

Eine Entscheidung für eine Vorgehensweise zu treffen, fällt schwer. Da der 3. Lösungsweg – welcher einen sequentiellen Ablauf aufweist – die geringsten Kosten verursacht und die Anwendung anhand des konkreten Beispiels ‚inSiDE‘ – erfolgt, wird dieser Lösungsweg auch tatsächlich in der konzeptionellen Handlungsempfehlung berücksichtigt. Ursächlich hierfür ist vor allem die Tatsache, dass ‚inSiDE‘ von einer öffentlichen Einrichtung publiziert wird und deshalb eine Kostenminimierung zwingend berücksichtigt werden muss. Ein weiterer Vorteil der Lösung für dieses Beispiel ist, dass der Autorenverbund, der einer der wichtigsten Bestandteile einer Publikation bildet, in seinem bisherigen Arbeitsablauf keinerlei Veränderungen erfährt und so in gewohnter Weise weiterarbeiten kann.

5 Umsetzung der evaluativ abgeleiteten Handlungsempfehlung

Das Ergebnis der vorangegangenen Evaluation ist eine konzeptionelle Handlungsempfehlung, die den Ist-Zustand (Status quo) in den Soll-Zustand (empfohlener Lösungsweg) überführen soll. Kapitel 5 behandelt nun die konkrete und operative Umsetzung der aus der Evaluation abgeleiteten Empfehlung für die Gestaltung und Ausrichtung des Publikations-Workflow. Eine solche Konkretisierung ist anhand eines praktischen Beispiels leichter und anschaulicher vorzunehmen, und wird daher auch insbesondere am Anwendungsfall ‚inSiDE‘ durchgeführt.

Bei der empfohlenen (und hier ausschließlich behandelten) Vorgehensweise handelt es sich um die sequentielle Prozessgestaltung. Die Inhalte werden über Word erstellt und als Word-Dokument an die DTP-Vorstufe übergeben. Das Druck-Produkt wird in InDesign aufbereitet; nach der Druckfreigabe werden die Daten in XML-Daten exportiert und automatisiert zum Web-Produkt aufbereitet.

Für eine erfolgreiche Umsetzung der konzeptionellen Handlungsempfehlung bedarf es weiterer Planungs- und Umsetzungsaktivitäten. Diese weiterführenden Überlegungen sind Gegenstand der folgenden Betrachtungen. Dabei wird die Umsetzung nicht vollständig und lückenlos beschrieben; vielmehr stellt der folgende Abschnitt eine Art Rezept dar, wie vorzugehen ist, um den empfohlenen Lösungsweg zu realisieren. Es soll also jeweils der Versuch unternommen werden, einige zentrale Eckpunkte einer konkreten Ausarbeitung herauszuarbeiten und zu benennen.

Die Handlungsempfehlung setzt sich aus den vier folgenden Maßnahmen zusammen:

1. Erzeugung einer geeigneten Datenstruktur
2. Wahl einer effizienten Speicher- und Verwaltungsmöglichkeit der Daten
3. Definition der Anweisungen zur Präsentation der Daten
4. Auswahl der notwendigen Komponenten, um eine Automatisierung zu ermöglichen.

Diese vier Einzelmaßnahmen werden im Folgenden näher beleuchtet.

5.1 Datenstruktur

Die Datenstruktur spiegelt sowohl den Aufbau eines einzelnen Artikels und einer Ausgabe der Publikation wider, als auch den vollständigen Zusammenhang der darin enthaltenen Datenelemente und den zwischen diesen bestehenden Beziehungen. Um eine möglichst wirkungsvolle und nachhaltige Strukturierung des Datenbestandes zu erreichen, ist deshalb zunächst eine Analyse der einbezogenen Inhalte zwingend erforderlich. Aus dem vorhandenen Datenbestand der älteren Printausgaben ergibt sich eine erste Aufbaustruktur der Ausgabe sowie der Artikel. Dies ist die Basis für die Konzeption einer Strukturdatei, beispielsweise einer DTD. Aus dieser Strukturdatei erfolgt zudem die Erstellung der Autoren-Vorlage zur Inhaltsbeschaffung.

Ziel ist es also, eine Struktur zusammen mit einem Regelwerk zu entwerfen, auf deren Grundlage eine Vorlage für die Autoren erstellt werden kann und welche als dauerhaftes Gerüst für den vollständigen Datenbestand dienen kann. Durch diese Struktur werden die Elemente definiert und die Beziehungen zwischen den Elementen geregelt.

Strukturaufbau

Die Strukturinformation kann über eine DTD oder ein XML-Schema realisiert werden. Hierbei stellt sich die Frage, welche der beiden Möglichkeiten gewählt werden soll. Da das hierfür eingesetzte Layoutprogramm InDesign XML-Schemata nicht unterstützt, erfolgt der Entwurf mittels einer DTD.⁶⁹

Die DTD stellt das Design der Daten vor. Da zahlreiche XML-Dokumente bereits vorliegen können und eine Vielzahl von Dokumenten im Publikationsverlauf neu hinzukommen wird, ist es zu empfehlen, diesen Umfang an Daten gezielt durch eine externe DTD als primäre Referenz für die XML-Dokumente zu steuern. Die DTD wird als separate Datei mit der Endung ‚.dtd‘ gespeichert. Für die Elemente, Attribute und Entitäten ist darüber hinaus ein Namenskonzept zu entwerfen, welches den allgemeinen Regeln der Namensvergabe genügt.⁷⁰ Bei abgekürzter Namensvergabe sollte für die bessere Verständlichkeit der volle Name in Kommentaren festgehalten werden. Die DTD dient als Regelwerk für eine Klasse von Dokumenten. Daher muss sie so flexibel und erweiterungsfähig entworfen sein, dass alle absehbaren Anforderungen dieser Dokumentenklasse erfüllt werden können. Der Entwurf einer guten DTD setzt eine geeignete Datenanalyse zur Vorbereitung voraus. Hier ist immer wieder das Augenmerk auf den Aufbau der Datenstrukturen zu richten. Die Analyse soll den Input für das Daten-Design liefern; je besser also grundsätzlich diese Analyse ist, umso leistungsfähiger ist die daraus gewonnene Datenstruktur. Dabei ist allerdings stets zu berücksichtigen, dass die Daten sich in Zukunft verändern können und wie diese Veränderungen ggf. in der DTD nachträglich integriert werden können.⁷¹

Die Entwicklung einer DTD lässt sich somit in verschiedene Stufen einteilen:

- Welche Ziele sind für das Projekt definiert?
- Welche Anforderungen bestehen an die Dokumente?
- Welche Anforderungen ergeben sich dadurch an die DTD? (Wie sieht die Datenstruktur aus, welche Daten sind zwingend erforderlich, welche nur optional, wie ist die Reihenfolge der Struktur, welche Elemente und Attribute werden welchen Inhalten zugeordnet?)
- Feinkonzept der DTD (Elemente auswählen, Hierarchie bilden, Validierung)
- Abschluss der Entwicklung und Implementierung der DTD
- Testing

⁶⁹ Eine ausführlichere Darstellung von DTD und XML-Schema wird beispielsweise in Klettke/Meyer (2003) geboten.

⁷⁰ Vgl. SELFHTML (2005a)

⁷¹ Vgl. SELFHTML (2005b)

Nach der Entwicklungsphase sind Testläufe notwendig. Ist das Konzept einer DTD erstellt und funktionstüchtig, ist anhand dieser die Word-Vorlage zu erstellen. Dabei dürfen nur die Elemente, die der Autor bei der Verfassung seines Artikels auch benötigt, in der Vorlage definiert sein.

Artikelanalyse

Eine Analyse bereits erstellter Artikel zeigt, dass die Autoren zwar oft eine Formatvorlage nutzen, aber immer wieder für einzelne Formatierungen auf die Schaltflächen, welche in Word zur Verfügung gestellt werden, zurückgreifen. Dies ist besonders nachteilig für die Weiterverarbeitung in der DTP-Vorstufe, da Auszeichnungen über diese Schaltfläche nicht erkannt werden. Den Autoren muss also deutlich gemacht werden, dass sie die Formatierung ausschließlich über die zur Verfügung gestellte Vorlage vornehmen dürfen. Weitere Besonderheiten fallen bei der Bildintegration auf; Bilder werden verknüpft oder separat übergeben, wobei dazu eine Bildmarke im Text-Dokument gesetzt wird. Oftmals werden Bilder auch eingebettet, wodurch die Qualität des Bildes leidet. Daraus kann geschlossen werden, dass den Autoren neben der eigentlichen Vorlage zur Formatierung der Artikelelemente eine Anleitung zur Nutzung dieser Vorlage bereitgestellt werden muss.

Die Dokumente sind aus Texten und Bildern zusammengesetzt, wobei Formeln dabei entweder als Zeichen im Text integriert oder als Bild enthalten sind. Aufgrund der Analyse der bereits verwerteten Artikel können folgende Elemente als wesentliche Bestandteile ermittelt werden:

- Autor (Name, Institut, Stadt, Land)
- Artikeltitle
- Unterüberschrift (bzw. Absatzüberschrift)
- Text
- Bild mit Bildunterschrift (teilweise finden sich zusätzliche Bildhinweise im Text)
- Formel mit Hinweis im Text
- Quellennachweis (Referenzen)

Periodikum-Analyse

Die Analyse zum Aufbau eines Periodikums kann anhand einer Druckversion vollzogen werden. Dabei können die vorhandenen Absatz- und Zeichenformate Aufschluss über die formatierten Elemente geben, die in der Artikelanalyse schon aufgezählt wurden. Die für eine DTD wichtigen Elemente (zusätzlich zu denen der Artikelanalyse) werden im Folgenden kurz dargestellt. Die Bezeichnungen werden dabei in englischer Sprache verfasst, da diese standardmäßig in der Fachzeitschrift ‚inSiDE‘ verwendet wird:

- First Page (Titelseite)
- Editorial (Vorwort)
- Contents (Inhaltsverzeichnis)
- Article (1-n)
- Publisher (drei Kurzportraits der kooperierenden Rechenzentren)

- Events (Kursdaten)
- Miscellany (Vermischtes)
- Impressum

Auf der Grundlage dieser beiden Analyseergebnisse ist nun das Konzept der DTD zu erstellen. Dabei müssen nicht nur die notwendigen Elemente integriert, sondern es müssen auch die Beziehungen zwischen den Datenelementen und die Attribute der Elemente eindeutig beschrieben werden. Unter Berücksichtigung dieser Aspekte lässt sich ein Entwurf für ein umfangreiches Regelwerk erstellen und in einer DTD festhalten. In Abbildung 21 ist der Auszug eines solchen Entwurfs, welcher in der Namensgebung an das Anwendungsbeispiel angepasst ist und hier einen kleinen Einblick bieten soll, dargestellt.

```
<!ELEMENT edition (firstpage, editorial, contents, article+, centers+, courses, miscellany, impressum)>
<!ELEMENT firstpage (volume, image)>
<!ELEMENT editorial (title, body, publisher)>
<!ELEMENT contents (title+)>
<!ELEMENT article (author, title, body, contact)>
<!ELEMENT centers (centername, body, contact, image, table)>
<!ELEMENT courses (centername, course+)>
<!ELEMENT miscellany (shortarticle)+>
<!ELEMENT impressum (publisher, editor, designer, author+, abo, copyright)>
...
```

Abb. 21: Auszug des DTD-Entwurfs der Fachzeitschrift ‚inSiDE‘

Aus den vorhandenen Elementen und Attributen werden die für eine Artikelformatierung wesentlichen ausgewählt und analog dazu die passenden Formate in der Vorlage für die Autoren erstellt. Diese sollten wiederum mit den Absatz- und Zeichenformaten der Layout-Datei übereinstimmen, um einen effizienten Textimport in das Layout-Programm zu gewährleisten. Folgende Elemente sollten für einen effektiven Einsatz einer Vorlage für Autoren ausgewählt werden:

- author (name, surname, department, university, country)
- title
- subtitle
- text
- equation
- sub
- sup
- image description

Alle Zeichen, die in einem Word-Dokument nicht gekennzeichnet sind, können nach dem Import in InDesign nicht den entsprechenden Absatz- bzw. Zeichenformaten zugewiesen werden. Elemente wie ‚Publisher‘, ‚Events‘ und ‚Impressum‘ werden nicht von den Autoren verfasst und müssen daher nicht in der Vorlage definiert werden.

Die weitere Vorgehensweise sieht die vollständige Entwicklung einer gültigen DTD vor. Diese kann über eine Software erfolgen, die eine Erstellung von DTDs ermöglicht, oder sie kann manuell über einen Texteditor eingegeben werden. Die Erstellung der Vorlage sollte direkt in Word vorgenommen werden, da sie anschließend in Word wieder Verwendung finden soll. Interessenten, die einen Artikel in der InDesign veröffentlichen wollen, erhalten auf Anfrage die erstellte Word-Vorlage sowie eine kurze Anleitung zur Artikelerstellung. Darin enthalten ist die Abgabefrist, die für die kommende Ausgabe gesetzt werden muss, damit diese pünktlich erscheinen kann.

5.2 Speicherung und Verwaltung von Daten

Eine sinnvolle Verwaltung von Inhalten bietet die Möglichkeit diese nach gegebenen Richtlinien zu ordnen, zu speichern und wiederzufinden, um sie weiter zu verwerten. Eine solche Verwaltung kann über den Einsatz eines Datenhaltungssystems geregelt werden. Dafür ist ein Konzept zu erstellen, welches alle Elemente der Daten erfasst und in Abhängigkeiten zueinander ordnen kann. Welche Art einer Datenbank am sinnvollsten zum Einsatz kommt oder ob die Ablage in einem einfachen Dateisystem ausreicht, sollte in einer detaillierten Ausarbeitung unter Berücksichtigung der gegebenen Anforderungen ermittelt werden.

Bei den zu erstellenden Dokumenten für die Fachzeitschrift ‚inSiDE‘ handelt es sich um dokumentenzentrierte XML-Dokumente, deren konzeptioneller Entwurf – wie bereits durch die Entwicklung eines DTD-Entwurfes erfolgt (vgl. Kapitel 5.1) – eine Modellierung von Inhalt und Struktur berücksichtigen sollte. Anfragen beziehen sich auf diese Strukturen und Inhalte und lassen sich mittels XML Path Language (XPath) realisieren. Änderungen können über geeignete Operationen im Document Object Model (DOM) vorgenommen werden. Auf physischer Ebene eignet sich eine Speicherung der Dokumentkollektionen als Dateien und eine Indizierung über diesen Dateien, wobei die Inhalte der Dokumente als Volltextindex und die Struktur als Struktur- und Pfadindex zu indizieren sind.⁷²

Prinzipiell kommen drei verschiedenen Möglichkeiten von Datenhaltungssystemen bei dem vorliegenden Anwendungsfall in Frage:

- ein Datei-basiertes System
- eine XML-erweiterte relationale Datenbank oder ein hybrides XML-Datenbanksystem⁷³
- eine native XML-Datenbank

⁷² XML-Dokumente lassen sich in dokumentenzentrierte, datenzentrierte oder semistrukturierte (Mischform) Dokumente einteilen. Eine Erläuterung dazu und zu weiteren Eigenschaften und Anforderungen an eine Datenbankarchitektur findet sich in Klettke/Meyer (2003), S. 100 ff.

⁷³ Dazu schreibt Gorke: „Da es für hybride Systeme keinen richtigen Standard gibt, ist die Grenze zwischen nur erweitertem System und hybriden System sehr schwammig. Der Trend geht auf jeden Fall zu den hybriden Systemen [...]“ Zitat aus Gorke (2006), S. 22. Gorke gibt außerdem an, dass hybride Systeme sich vor allem in der Einbindung einer nativen Zugriffsschicht zu erweiterten Systemen unterscheiden. Diese bietet direkten Zugriff auf XML-Daten mittels XQuery. Vgl. Gorke (2006), S. 21 ff

Die Systeme können als kleine Mediendatenbanken betrachtet werden, die über ein User-Frontend, einem Medien-Application-Server und je nach Verwaltungswahl einem File- oder Datenbank-Server in etwa dem klassischen Aufbau einer Mediendatenbank entsprechen. Jedes System hat seine spezifischen Vor- und Nachteile, auf die hier jedoch nicht näher eingegangen werden soll. Der Bedarf eines Verwaltungssystems ist im vorliegenden Anwendungsfall derzeit nicht vorhanden. Durch zukünftige Ausgaben der ‚inSiDE‘ vergrößert sich aber die Anzahl der Dateien, besonders die der Bilddaten, immer weiter; diese Entwicklung sollte bei der Entscheidung darüber, welches System letztlich gewählt wird, berücksichtigt werden.

Die weitere Vorgehensempfehlung zu der hier aufgeführten Maßnahme beinhaltet die Wahl eines Datenhaltungssystems, welches den genannten Anforderungen entspricht, und die Erstellung eines Datenbankkonzeptes, das sich an dem Entwurf der DTD orientiert.

5.3 Darstellung der Daten

Die vorhandenen und entstandenen Inhalte sollen einerseits als Druckerzeugnis und andererseits als Web-Produkte dargestellt werden. Um die geforderte Darstellung der Inhalte zu ermöglichen, müssen den vorhandenen Elementen Gestaltungsangaben zugewiesen werden. Diese werden über Stylesheets definiert, welche wiederum zur Präsentation von Daten dienen. Für das Druckprodukt gelten andere Gestaltungsregeln als für das Web-Produkt, somit müssen zwei unterschiedliche Regelwerke, die sich jedoch auf dieselben Datenelemente beziehen, erstellt werden; ein Print Stylesheet (s. Abbildung 22), das in InDesign durch Absatz- und Zeichenformate definiert wird und ein Web Stylesheet (s. Abbildung 23), das über die Stylesheet-Sprachen XSLT und/oder CSS beschrieben werden kann.⁷⁴



Abb. 22: Screenshot eines Print Stylesheet in Form von Absatzformaten

⁷⁴ Vgl. Kapitel 3.4.1


```
.orange{ background-color : #F0D191;}
.oro{ color : #DBA64F;}
.hell{ background-color : #F8E8C7;}

.headline{
    font-family: Arial, Helvetica, sans-serif;
    font-size : 14pt;
    text-align : left;
    background-color : #F8E8C7;}
...
```

Abb. 23: Ausschnitt aus einem Web Stylesheet in CSS-Form

5.4 Automatisierung des Web Workflow

Durch eine Automatisierung sollen technische Vorgänge ohne menschlichen Einfluss selbsttätig erfolgen. Im Beispiel dieser Arbeit ist damit der Publikationsprozess der Online-Ausgabe gemeint. Welchen Grad die Automatisierung dabei erreicht, ist von der eingesetzten Technologie abhängig. Laut Knobloch und Kopp (2001) ist der erste Schritt in Richtung automatisierte Informationsproduktion die Möglichkeit, Inhaltsdaten und ihre Gestaltung mit Hilfe von Transformationen zu vereinen.⁷⁵

Aus den exportierten XML-Daten kann über einen entsprechenden Transformationsvorgang eine Online-Ausgabe automatisiert erzeugt und dargestellt werden. Dafür wird ein Transformationsprozessor eingesetzt, welcher die Aufgabe hat, die XML-Elemente in HTML-Elemente umzuwandeln. Die notwendigen Steueranweisungen werden in Form von Regeln mittels XSLT beschrieben.⁷⁶ Das Regelwerk muss entsprechend der vorhandenen Strukturen (veranschaulicht über eine DTD) und gewollten Darstellung (Web-Präsentation) der Daten angefertigt werden. Der Prozessor baut zum Verarbeiten eines Quelldokuments (XML) eine hierarchische Struktur aus Knoten auf, auf die er die Steueranweisungen aus der XSLT-Datei anwendet. Er bildet aus Quelldokument und Informationen der XSLT-Datei eine neue Informationseinheit, die das gewünschte Zieldokument (HTML) darstellt.

Transformationen dienen folglich zum Konvertieren der Daten von einem Format in ein anderes. Um die Automatisierung vollständig gewährleisten zu können, kann die Generierung der HTML Seiten on-the-fly mit einem Batchjob erfolgen. Der Batchjob kann zu einem gewählten Zeitpunkt, beispielsweise ein Tag vor dem Publikationszeitpunkt der gedruckten ‚inSiDE‘-Ausgabe, aus der XML-Datei die HTML-Dateien generieren. Zu den Komponenten, die für einen erfolgreichen Transformationsprozess zur Präsentation der Daten als Web-Produkt erforderlich sind, zählen somit:

- ein Transformationsprozessor
- ein oder mehrere Stylesheets für die Transformationsanweisungen in XSLT
- eine Batchjob-Programmierung

⁷⁵ Vgl. Knobloch/Kopp (2001), S. 12

⁷⁶ Vgl. Kapitel 3.4.1

Konkret übertragen auf den vorliegenden Anwendungsfall entstehen folgende Anforderungen an den Transformationsprozess:

- Es sollen mit den vorhandenen Artikeln <article> ein Inhaltsverzeichnis mit den Titeln <title> erstellt werden, so dass die Angaben im Inhaltsverzeichnis auf die einzelnen Artikel verweisen. Zu den Artikeln gehört auch das Editorial, das jeder neu erscheinenden Ausgabe beigelegt ist.
- Des Weiteren soll die neue Ausgabe als aktuelle Ausgabe erscheinen und die bisher aktuelle Ausgabe unter dem Menüpunkt ‚Archiv‘ auf der Website erscheinen.
- Sonstige Inhalte wie ‚Centers‘, ‚Events‘ oder ‚Impressum‘ müssen nur gegebenenfalls in speziellen Abschnitten aktualisiert werden. Hierbei wird deutlich, dass die schon vorhandene HTML-Publikation wieder verwendet werden kann und nur die entsprechenden Inhalte, die aktualisiert werden müssen, einfließen müssen.

Die folgende Abbildung zeigt zur Veranschaulichung einen Screenshot der aktuellen Online-Ausgabe der ‚inSiDE‘.



Abb. 24: Screenshot der Online-Ausgabe ‚inSiDE‘, <http://inside.hlr.de>

Es soll eine XSLT-Datei entworfen werden, welche die Inhalte des neu erzeugten XML-Dokuments mit dem bestehenden HTML-Gerüst verbindet. Dazu werden die aktuellen Inhalte aus dem XML-Dokument durch den Prozessor ausgelesen und über XSLT-Anweisungen, die mit entsprechenden HTML-Elementen in der XSLT-Datei verbunden sind, in eine neu generierte HTML-Datei

eingebaut. Sowohl die Transformation als auch die Darstellung der Web-Daten erfolgt jeweils über die enthaltenen Informationen einer XSLT-Datei. Die erzeugte HTML-Datei kann jedoch durch weitere Gestaltungsanweisungen wie Schriftart, -farbe, Hintergrundbild etc. über eine eingebundene CSS-Datei ergänzt werden. Über CSS kann die Positionierung der Elemente etwas freier gestaltet und die Lesbarkeit bei umfangreichen Dokumenten erhöht werden. Letztendlich soll eine sinnvolle Kombination aus XSLT und CSS gefunden werden, wobei CSS eingesetzt werden sollte, wenn es ausreichend und XSLT, wenn es nicht anders möglich ist.

Im weiteren Verlauf der Planung sollten nun die notwendigen Komponenten ausgewählt, ggf. angepasst und eingesetzt werden. Für den Transformationsvorgang kann eine bereits am Markt vorhandene kommerzielle oder Open Source Lösung verwendet und an die gegebenen Anforderungen angepasst werden, wodurch eine Neuentwicklung vermieden werden kann. Der Prozess könnte beispielsweise über den XSLT-Prozessor ‚XALAN‘ ablaufen.

6 Fazit und Ausblick

„Die Arbeit mit elektronischen Dokumenten aller Art steht vor einem historischen Umbruch. Die Zeit der alten Microsoft-Formate wie doc (für Texte) oder xls (für Tabellen) läuft ab. Die Zukunft gehört dem XML-Standard, der zwei große Vorteile hat: Erstens wird dieser nicht von einer einzigen Firma kontrolliert und zweitens kann er im Unterschied zu den alten binären Formaten von allen gelesen werden, benötigt dafür also keine besondere Software.“⁷⁷

Der XML-Standard hat sich zu dem zentralen Austauschformat für die Gestaltung und Organisation von cross-medialen Publikationsprozessen entwickelt. In der vorliegenden Arbeit wurden Lösungswege zur cross-medialen Umsetzung wissenschaftlicher Texte untersucht. Unter den gegebenen Anforderungen einer kleineren wissenschaftlichen Publikation (,inSiDE') wurden eingehend verschiedene Lösungsmöglichkeiten zur Erstellung, Übergabe und Verarbeitung der Publikationsinhalte behandelt. Ein Schwerpunkt der Untersuchung lag dabei auf dem Umgang mit den in den Publikationsinhalten eingebetteten mathematischen Ausdrücken in der jeweiligen Phase des Workflow. Die aus dieser Analyse erarbeitete Evaluation der alternativen Lösungsansätze endet schließlich in einer detaillierten konzeptionellen Handlungsempfehlung.

Das Resultat der Evaluation ist damit ein Gestaltungskonzept für einen sequentiell organisierten Verarbeitungsprozess mit Integration eines automatisierten Web Workflow. Mathematische Ausdrücke werden dabei in Bildform innerhalb des Publikationsprozesses verarbeitet.

Zusammenfassend lässt sich feststellen, dass der empfohlene Lösungsweg erste wichtige Verbesserungen des hier vorgestellten Anwendungsbeispiels ,inSiDE' hervorbringen kann. Im Rahmen der durch die Ausgangslage vorgegebenen Lösungsbedingungen – hier ist besonders die weitgehende Festlegung der Software-Landschaft anzuführen – wird mit der Umsetzung der Handlungsempfehlung eine deutliche Verbesserung der Effizienz des Publikationsprozesses konkret erreichbar. Wird diese pragmatische Grundausrichtung des Workflow-Konzeptes in Form einer engen Anlehnung an die bestehenden Arbeitswerkzeuge allerdings fallen gelassen, sind auch andere, weitgreifendere Lösungsoptionen möglich. Dies war aber von vornherein nicht Gegenstand der Arbeit.

Aber auch innerhalb des hier empfohlenen Lösungsansatzes existieren noch weitere Reserven für Verbesserungen. So kommt, wie in der Arbeit ausführlich dargelegt, der Auszeichnung in MathML eine wichtige Funktion für die Verarbeitung der Publikationsinhalte zu. Gerade für mathematische Ausdrücke stellt MathML ein attraktives Austauschformat zur Verfügung. Dies würde jedoch eine Umstellung der Arbeitswerkzeuge mit sich bringen oder erhöhten Aufwand durch die Einführungen eines webbasierten Content Management Systems – mit sinnvollen Erweiterungen zur Formelerstellung – erfordern.

⁷⁷ Zitat von Zschunke (2006)

Somit ist festzuhalten, dass das vorliegende Ergebnis der Evaluation eine geeignete und fundierte sowie die unter den gegebenen Bedingungen beste Lösung für den gesuchten Workflow darstellt. Gleichwohl ist zu beachten, dass sich dieser Softwarebereich gegenwärtig stark im Umbruch befindet. Es ist anzunehmen, dass sich die bereits begonnene Entwicklung fortsetzt und die Software-Anwendungen um Schnittstellen für weitere XML-Spezifikationen wie MathML erweitert werden; es gilt also, diesen Entwicklungsverlauf weiterhin aufmerksam zu verfolgen. Neue Entwicklungen erfordern dann allerdings eine neue Evaluation der Handlungsmöglichkeiten.

Eine Folge des technologischen Wandels, die hier abschließend erwähnt sei, ist die Ankündigung von Microsoft, 2007 ein neues Office Paket mit dem Datenformat ‚Open XML‘ auf den Markt zu bringen. Es bleibt abzuwarten, ob dieses neue Format auch tatsächlich eine reibungslose Integration in eine bestehende XML-Umgebung ermöglicht und ob die angestrebte Abbildung von Formeln gelingt; eines wird jedenfalls deutlich: in eine zentrale Fragestellung der Dokumentenverarbeitung ist Bewegung gekommen.

Literaturverzeichnis

Büchner, H., Zschau, O., Traub, D., Zahradka, R. (2001)

Web Content Management,
Galileo Press GmbH

Forssman, F., de Jong, R. (2002)

Detailtypografie,
Hermann Schmidt Verlag

Gorke, B. (2006)

XML-Datenbanken in der Praxis,
bomots verlag

Heinrich, L., Häntschel, I. (2000)

Evaluation und Evaluationsforschung in der Wirtschaftsinformatik,
Oldenbourg Wissenschaftsverlag GmbH

Heinrich, L., Lehner, F. (2005)

Informationsmanagement
8. Auflage, Oldenbourg Wissenschaftsverlag GmbH

Klettke, M., Meyer, H. (2003)

XML & Datenbanken, 1. Auflage
dpunkt.verlag GmbH

Knobloch, M., Kopp, M. (2001)

Web-Design mit XML, 1. Auflage
dpunkt.verlag GmbH

Kretzschmar, O., Dreyer, R. (2004)

Medien-Datenbanken und Medien-Logistik-Systeme,
Oldenbourg Wissenschaftsverlag GmbH

Meister, C., Gahler, T., Jamieson, P., Freßdorf, C. (2006)

Microsoft Word-Programmierung – Das Handbuch,
Microsoft Press Deutschland

Rothfuss, G., Ried, C. (2003)

Content Management mit XML,
Springer Verlag

Sandhu, P. (2003)

The MathML Handbook,
Charles River Media Inc.

Elektronisches Quellenverzeichnis

Contentmanager.de (2006)

<http://www.contentmanager.de/itguide/produktfinder.html>, Abruf vom 20.7.2006

Heise-Online (2006)

<http://www.heise.de/glossar/entry/7f73f4589af5483c>, Abruf zum 11.7.2006

HLRS (2006)

<http://www.hlrs.de>, Abruf vom 30.6.2006

Marti, F. (2006)

Cross Media Publishing – integrierte Lösung fürs Marketing, 04/2006,

http://www.contentmanager.de/magazin/artikel_971_cross_media_publishing_integrierte_loesung.html, Abruf vom 30.6.2006

Mathe Online (2004)

<http://www.mathe-online.at/formeln.html>, Abruf vom 17.4.2006

Nix, M. (2004)

Generation XML: XML als universelle Auszeichnungssprache, 08/2004 ,

http://www.contentmanager.de/magazin/artikel_564_xml_als_universelle_auszeichnungssprache.html, Abruf vom 30.6.2006

SELFHTML (2005a)

<http://de.selfhtml.org/xml/dtd/bearbeitungsregeln.htm#namen>, Abruf vom 25.7.2006

SELFHTML (2005b)

<http://de.selfhtml.org/xml/dtd/allgemeines.htm>, Abruf vom 30.6.2006

Spiegel Online (2006)

Lachsrosa Redaktion für Print und Online, 12.7.2006,

<http://www.spiegel.de/netzwelt/netzkultur/0,1518,426363,00.html>, Abruf vom 12.7.2006

Quality-Datenbank (2006)

<http://www.quality.de/lexikon/gebrauchstauglichkeit.htm>, Abruf zum 23.7.2006

W3C (2006)

<http://www.w3.org/Math/>, Abruf vom 13.7.2006

Zschunke, P. (2006)

Streit um die Revolution, (23.7.2006),

<http://www.spiegel.de/netzwelt/politik/0,1518,428010,00.html>, Abruf zum 23.7.2006