Torsten Schaßan

Digitale Quellen: Datei- und Datenformate

1. Geschichtswissenschaft und digitale Quellen

1.1 Einleitung

Digitale Quellen sind Teil von Forschung und Lehre und sind wie analoge Quellen im Sinne der Quellenkritik grundsätzlich auf folgende Kriterien hin zu überprüfen:

- Authentizität der Daten: Welche Herkunft haben die Quellen? Sind die Quellen authentisch bzw. autorisiert?
- Entstehungsbedingungen der Quellen: Mit welchen Medientechniken wurde gearbeitet? Wie sind die Quellen strukturiert? Ist die Struktur der Information angemessen? Welche Qualitätsmaßstäbe wurden bei der Erstellung der Daten angewendet?
- Rechtsstatus der Quellen: Gibt es rechtliche Beschränkungen in der Nutzung der Quellen?

Im Fall digitaler Quellen sind neben diesen Kriterien der klassischen Quellenkritik und ihren methodischen Fragen aber auch technische Aspekte einzubeziehen. Die Formate der Daten sowie deren Vergangenheit (die Entstehung der Daten) als auch deren Zukunft (intendierte oder geplante Verwendung der Daten) spielen eine Rolle. Die Eigenschaften der Dateiformate bestimmen die Möglichkeiten der Weiterverarbeitung, wie auch Integrität und Authentizität der Daten dadurch determiniert sein können. Daher rücken zwei weitere Kriterien in den Fokus einer digitalen Quellenkritik: Erstens das (technische) Format der Daten: In welchen Formaten liegen die Quellen vor? Zweitens die Weiterverarbeitungsmöglichkeiten: Welche Möglichkeiten der Auswertung, Be- und Weiterverarbeitung bieten diese Formate?

In diesem Clio-Guide werden die beiden letztgenannten technischen, nur bei digitalen Quellen relevanten Fragestellungen untersucht. Es werden zunächst grundlegende Gemeinsamkeiten von Dateiformaten herausgearbeitet, wobei allgemeine Überle-

http://www.clio-online.de/guides/arbeitsformen-und-techniken/digitale-quellen-dateiformate/2018

gungen zur Authentizität von Daten vorangestellt werden sollen. Danach werden die gängigsten Dateiformate für Text- und Bildquellen beschrieben und auf Entstehung und Möglichkeiten der Weiterverarbeitung hin untersucht. Da Texte und Bilder immer noch die zentralen historischen Quellen sind, werden Audio- und Videoformate nicht eigens beschrieben. Für Text-basierte Daten wird zudem eine Darstellung wichtiger Datenformate geliefert, wobei der Schwerpunkt auf die Text Encoding Initiative (TEI) gelegt wird. Dabei werden auch die unterschiedlichen Entstehungsbedingungen und deren Einfluss auf Struktur und Qualität der Daten thematisiert. Abschließend werden als Ausblick knapp Möglichkeiten des Semantic Web skizziert.

1.2 Authentizität und Integrität digitaler Quellen

Die Zuschreibung von Authentizität erfolgt im Digitalen zunächst wie im Analogen durch Nennung der Autorschaft. Der oder die UrheberIn bzw. der Weg der Erstellung von Daten muss durch entsprechenden Angaben bzw. beschreibenden Daten (*Metadaten*) erkennbar werden: In allen Dateiformaten können AutorInnen bzw. ErstellerInnen sich als AutorIn im Text/Bild/Ton nennen. Der Name wird dann auf dem Titelblatt stehen oder zu Beginn der Datei genannt. Diese Daten sind für menschliche BenutzerInnen sichtbar und können dann auf ihre Authentizität hin überprüft werden. Eine maschinelle Auswertung ist auf diesem Wege eher schwierig.

Für die maschinelle Verarbeitung der Metadaten gibt es in nahezu allen Datenformaten spezielle Eigenschaften, in denen die Metadaten eingetragen werden können und eingetragen sein sollten. In den gängigen Text- und Bildformaten werden Metadaten allerdings auch oft automatisiert erstellt. Office-Installationen können beispielsweise während der Installation durch Angabe des eigenen Namens personalisiert werden, so dass alle auf dem entsprechenden Computer erstellte Dokumente mit Angaben zur Autorschaft (Name) der versehen werden. Erfolgt allerdings während der Softwareinstallation nur eine unvollständige Identifizierung, werden diese unvollständigen Angaben in alle Dokumente übernommen. Wenn der oder AutorIn eines Dokumentes die all-

gemeinen Einstellungen nicht ergänzt oder korrigiert, wird die Zuschreibung der Autorschaft entsprechend schwierig.

In der vernetzten Welt kann die Integrität von Daten auch auf dem Übertragungsweg leiden: Einerseits können in der Übertragung einzelne oder mehr Bits der Daten fehlerhaft übertragen und die Daten damit korrumpiert werden. Andererseits könnten die Daten durch Dritte abgefangen, verändert und danach erst zugestellt werden. Besonders bei rechtsverbindlichen Dokumenten muss sichergestellt werden, dass der angenommene UrheberIn dem tatsächlichen entspricht und Dokumente nicht durch Dritte verändert wurden.

Um den möglichen Fall einer technischen Störung bei der Übertragung zu überprüfen, wird für viele Daten ein sogenannter *Hashwert*[1] bzw. eine Prüfsumme berechnet, der gemeinsam mit den Daten veröffentlicht wird. Der oder die EmpfängerIn der Daten kann seinerseits den Hashwert der empfangenen Daten berechnen und mit dem der zur Verfügung gestellten Daten vergleichen. Sind beide Werte identisch, kann man davon ausgehen, dass die gesendeten Daten integer sind.[2] Im Fall der absichtlichen Veränderung von Daten während einer Übertragung reicht diese digitale Signatur nicht aus. Bei einer sogenannten *Man-in-the-Middle-Attacke*[3] ist die abgesendete Datei möglicherweise digital signiert oder mit einer Prüfsumme versehen, dennoch könnten Daten verändert worden sein. Um dies zu vermeiden, kommt nur die komplette Verschlüsselung des gesamten Übertragungskanals in Betracht.

---

[1] https://de.wikipedia.org/wiki/Hashfunktion

[2] In der Vergangenheit wurde dazu vor allem der MD5-Wert berechnet. Diese Hashfunktion gilt inzwischen nicht mehr als sicher, da es mit überschaubarem Aufwand möglich ist, unterschiedliche Nachrichten zu erzeugen, die den gleichen MD5-Hashwert aufweisen. Derzeit ist MD5 nur bezüglich der Kollisions-Angriffe gebrochen. Deswegen besteht noch keine akute Gefahr für Passwörter, die als MD5-Hash gespeichert wurden. Diese Kollisionen sind eher eine Gefahr für digitale Signaturen, vgl. https://de.wikipedia.org/wiki/Message-Digest_Algorithm_5.

[3] https://de.wikipedia.org/wiki/Man-in-the-Middle-Angriff

Für solch eine Verschlüsselung von Daten und Kommunikation werden verschiedene Techniken angewendet. Um die Kommunikation mit Webseiten, die Eingabe von Daten oder das Abrufen von Emails sicherer zu machen, muss beispielsweise das *Übertragungsprotokoll*[4] gewechselt werden: Statt Webseiten mit dem Präfix „http://“ aufzurufen, muss man „https://“ davor setzen. https steht für das *Hypertext Transfer Protocol Secure*. Mit diesem Protokoll werden Daten werden beim Transport zunächst mittels eines standardisierten Verfahrens verschlüsselt (SSL/TLS - *Secure Socket Layer*/*Transport Layer Security*), dann wie gewohnt zum Zielort transportiert und dort wieder entschlüsselt. Bei Kommunikation, die nicht auf unmittelbarem Austausch beruht, sondern Nachrichten nur versendet und gegebenenfalls später erst „gelesen“ werden, wie beim Email-Verkehr, muss die Verschlüsselung anders organisiert werden: Hier beruht sie zum Beispiel auf einem *Public-Key*-Verfahren, in dem sich der oder die EmpfängerIn einen Schlüssel generiert, der aus einem öffentlichen und einem privaten Teil besteht. Den öffentlichen Teil eines Schlüssels dürfen alle NutzerInnen kennen und damit Nachrichten für die Empfängerseite verschlüsseln, auf der diese wiederum mittels öffentlichem Schlüssel dechiffriert werden. Der private Teil ist nur dem oder der EmpfängerIn bekannt und durch ein Passwort geschützt. Wahlweise kann man die Nachricht nur signieren oder komplett verschlüsseln. Die bekannteste Implementation der Public-Key-Technik ist OpenPGP, *Open Pretty Good Privacy*.

2. Datei- und Datenformate digitaler Quellen

2.1 Dateiformate

Die in diesem Guide beschriebenen Dateiformate lassen sich in vereinfacht in zwei Kategorien einteilen: Text-basierte und Bild-hafte. Es existieren aber auch Formate wie XML, das zwar ein Text-basiertes Format ist, gleichzeitig aber auch Eigenschaften einer Datenbank aufweist. Auch in Bild-haften Formaten wie SVG

[4] Das Übertragungsprotokoll beschreibt den Ablauf der Kommunikation und der Übertragung von Daten zwischen Sender und Empfänger.

verschwimmen die Grenzen, da bei SVG die Möglichkeit besteht, Text mit abzuspeichern und durchsuchbar zu machen.

Im Folgenden werden aus diesen zwei Kategorien nur die wichtigsten Formate näher betrachtet und deren Eigenschaften beispielhaft beleuchtet. Ausgeschlossen von der Betrachtung sind beispielsweise Archivformate wie .zip, sowie bestimmte Office-Formate wie Tabellen- oder Präsentationsformate (.xls, .ppt, usw.)

Text-basierte Formate

Bei den Text-basierten Formaten müssen reine Textformate (Dateiendung zum Beispiel .txt), Textformate mit Auszeichnung (zum Beispiel HTML- oder XML-Dateien) sowie komplexere Formate unterschieden werden, die unter anderem auf Formatierung abzielende Kodierungen enthalten (Dateien im Format .doc/.docx/.odt oder .pdf).

Reine Textformate

Bei einer reinen Textdatei, auch *plain text* genannt, wird der Inhalt als sequenzielle Folge von Zeichen eines Zeichensatzes[5] interpretiert. Die einzige Möglichkeit zur Gliederung eines Textes besteht in der Verwendung von Steuerzeichen wie Zeilenwechseln bzw. der besonderen Verwendung von Leerzeichen. Reine Textdateien enthalten zudem keinerlei Angaben zur Formatierung eines Textes. Dies steht im Gegensatz zu Binärdateien, bei denen eine beliebige anderweitige Interpretation des Inhalts möglich ist. Folglich ist eine ***Textdatei***[6] im Gegensatz zu einer Binärdatei ohne die Verwendung spezieller Programme lesbar und kann mit jedem Texteditor betrachtet und bearbeitet werden. Reine Textformate enthalten keine Metadaten in separaten Headern, sondern speichern den Zeichenstrom unmittelbar in der gewählten Kodierung

[5] Zu unterscheiden ist der hier allgemein verwendete Begriff als eine Menge von Zeichen von der umgangssprachlich häufig anzutreffenden Verwendung des Begriffs „Zeichensatz“ im Sinne einer speziellen Schrifttype, zum Beispiel Arial oder Times New Roman.

[6] https://de.wikipedia.org/wiki/Textdatei

ab. Daher kommt der *Kodierung*[7] der Datei die zentrale Rolle zur Interpretation und Verarbeitung einer Textdatei zu. Die wichtigsten Kodierungen sind Unicode, ISO 8859-1 bzw. -15 und ASCII.

Als historisch älteste Variante bot ASCII die Möglichkeit, lediglich 128 (Verwendung von 7 Bit) verschiedene Zeichen zu repräsentieren. Aufgrund von Verbesserungen in der Signaltechnik konnte bald das achte Bit ebenfalls genutzt werden, woraus sich die Kodierung ISO 8859 entwickelte.[8] In der Kodierung ISO 8859 kommt der Wahl der Schrifttype eine übergroße Rolle zu: Ist eine Datei technisch in ISO 8859-1 kodiert, aber unter Verwendung eines griechischen Zeichensatzes abgespeichert worden, so dass der Text „aussieht" wie ein griechischer, muss auf dem Zielcomputer, auf dem die Datei wieder geöffnet und weiterverarbeitet werden soll, notwendigerweise die gleiche Schrifttype installiert und ausgewählt sein, damit die Zeichen wiederum das gleiche Aussehen und damit die gleiche „Bedeutung" haben. Ist dies nicht der Fall, greifen die Betriebssysteme auf Standardtypen zurück und verändern damit möglicherweise das Aussehen des Textes. Um diesem Mangel und dem Grund für willkürliche Darstellung von Texten Abhilfe zu schaffen, wurde *Unicode*[9] erfunden. Unicode basiert auf der Überlegung, jedem sinntragenden Schriftzeichen oder Textelement aller bekannten Schriftkulturen und Zeichensysteme einen digitalen Code(-punkt) zuzuweisen, damit jedes Zeichen eindeutig zu identifizieren und von allen anderen Zeichen zu unterscheiden. Unicode wird ständig um Zeichen weiterer Schriftsysteme ergänzt. Der vom Unicode-Standard beschriebene Bereich umfasst derzeit 1.114.112 Codepunkte, von denen aktuell über 120.000 in 129 Schriftsystemen belegt sind. Während Unicode aber nur eine allgemeine Idee repräsentiert, muss man davon die technische Implementation unterscheiden:

[7] https://de.wikipedia.org/wiki/Zeichenkodierung

[8] Vgl. https://de.wikipedia.org/wiki/ISO_8859. Je nach Sprachgebiet wird das achte Bit anders belegt. ISO 8859-1, mit Zeichen aus dem westeuropäischen Bereich, wird auch als Latin-1 bezeichnet.

[9] http://www.unicode.org

Die wichtigsten *Implementationen von Unicode*[10] sind UTF-8 und UTF-16.

Obwohl heute nahezu alle Texteditoren Unicode-fähig sind, muss dennoch beachtet werden, in welcher Kodierung neue Dateien angelegt werden: Betriebssysteme, vor allem Windows, sind häufig noch auf den westeuropäischen Standard ISO 8859-1 eingestellt, so dass neu erstellte Dateien meist in dieser Kodierung angelegt werden. Texteditoren wie zum Beispiel der *freie Editor NotePad++*[11] sind in der Lage, die Kodierung einer Datei anzuzeigen und insbesondere zwischen ISO 8859 und Unicode zu konvertieren. Wird eine Datei durch ein Programm, zum Beispiel einen Browser, in einer falschen Kodierung dargestellt, werden Zeichen, die über den kleinsten gemeinsamen Bereich, den ASCII-Zeichen, hinaus unterschiedlich kodiert sind, fehlerhaft dargestellt. Beispielsweise werden in Dateien, die eigentlich in Unicode kodiert sind aber wie eine ISO 8859-kodierte Datei angezeigt werden, das „ü“ als „Ã¼“ angezeigt, usw. Umgekehrt wird in einer Datei, die in ISO 8859-1 gespeichert ist, aber als Unicode angezeigt wird, das „ü“ nur als „�“ angezeigt.[12] Bei einer Webseite, deren Kodierung falsch interpretiert und daher fehlerhaft dargestellt wird, reicht es normalerweise, im Browser die richtige Kodierung auszuwählen. Bei einer Textverarbeitung hingegen muss man die Datei (möglichst ohne zu speichern) wieder schließen und versuchen, sie mit der richtigen Kodierung oder mit einem anderen Programm zu öffnen. Insbesondere Microsoft-Office-Programme bieten mit den Standardeinstellungen beim Öffnen von Dateien keinen Dialog an, in dem man die Kodierung auswählen bzw. festlegen kann.

---

[10] https://de.wikipedia.org/wiki/Unicode_Transformation_Format

[11] https://notepad-plus-plus.org/

[12] Für weitere Aspekte zur Zeichenkodierung im Zusammenhang mit historischen Dokumenten vgl. Roeder, Torsten, Alpha into Alif. Schnittstellen zwischen Schriftkunde und Informatik am Beispiel von Unicode im Glossarium Graeco-Arabicum, in: Studia graeco-arabica 5 (2015) S. 345–363, https://www.academia.edu/14639413.

Textformate mit Textauszeichnung (Markup)

Reine Textdateien können durch sogenannte Textauszeichnung (Markup) um Textstrukturen und/oder Informationen zur Darstellung des Textes ergänzt werden. *Auszeichnungssprachen*[13] kennzeichnen Teile („Elemente“) von Texten oder anderen Daten mit Klammern (tags), die Anfang bzw. Ende der jeweiligen Eigenschaft des Textes, markieren. Im Text können dadurch semantische Strukturen explizit gemacht werden. Die explizite Auszeichnung von Text mittels Elementen und ähnlichem wird ergänzt um eine Kurzschreibweise, die als Markdown bezeichnet wird. Diese wird vor allem für die Beschreibung des Layout und rudimentäre Textstrukturen genutzt und kommt vor allem intern in Datenbanksystemen wie Wikis zum Einsatz.

Die bekannteste Auszeichnungssprache ist HTML.[14] HTML dient zur Strukturierung digitaler Dokumente wie Texten mit Hyperlinks, Bildern und anderen Inhalten. Anders als bei reinen Textdateien kann die Erweiterung um die strukturierenden Tags dazu genutzt werden, zusätzliche Metainformationen zu transportieren. Dazu wird eine HTML-Datei in Kopf und Körper geteilt. Der Kopf enthält die Metainformation, während der Körper die darzustellenden Inhalte aufnimmt.

Während HTML von Beginn an als *beschreibende* Auszeichnungssprache angelegt war, gab es doch einige Elemente, welche die Grenze zur *darstellenden* Textauszeichnung überschritten. Diese sind nach und nach als „missbilligt“ gekennzeichnet und aus dem Standard gelöscht worden. Die Darstellung der semantischen Einheiten kann stattdessen über Stilinformationen unmittelbar in jedem einzelnen Element festgelegt werden. Der bevorzugte Weg ist allerdings, die Stilinformation einmalig global für alle Vorkommen eines Elementes zu definieren und diese Infor-

[13] https://de.wikipedia.org/wiki/Auszeichnungssprache

[14] HTML steht für Hypertext Markup Language. Die wichtigsten, heute noch weit verbreiteten Versionen sind: HTML 4.01 mit der Spezifikationen Strict, Frameset und Transitional; XHTML 1.0 und HTML5. Vgl. http://www.w3.org/TR/html und https://de.wikipedia.org/wiki/Hypertext_Markup_Language

mation gegebenenfalls auszulagern. Diese Technik bezeichnet man als *Cascading Style Sheets (CSS)*[15].

Neben die Trennung von Inhalten (HTML-Elemente) und Form (CSS-Anweisungen) treten immer häufiger prozedurale Elemente von Programmiersprachen wie JavaScript. *JavaScript*[16] ist eine Skriptsprache, die ursprünglich für dynamisches HTML in Webbrowsern entwickelt wurde, um Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren und so die Möglichkeiten von HTML und CSS zu erweitern. Beispielsweise ist es mit der AJAX-Technologie[17] möglich, einzelne bereits geladene Webinhalte gezielt zu ändern und nachzuladen, ohne dass die ganze Seite neu geladen werden muss. Auf vielen Webseiten wird die Vorschau auf Suchergebnisse während der Eingabe eines Suchbegriffs mit AJAX realisiert. Als eine vollwertige Programmiersprache ist JavaScript allerdings auch eine Gefahrenquelle für die BenutzerInnen. Es können nicht nur dem Benutzer ungewollte Programmverhalten aufgezwungen, sondern gegebenenfalls auch Schadsoftware auf dem Rechner ausgeführt werden. Browser bieten deshalb mittlerweile oft die Möglichkeit an, eingebettetes JavaScript nicht auszuführen. Allerdings sind dann zahlreiche Webseiten nur eingeschränkt funktional und manche nahezu unbrauchbar.

HTML ist als Auszeichnungssprache eine Anwendung der *Standard Generalized Markup Language* (SGML). SGML wurde bereits 1986 als ISO 8879:1986 standardisiert. SGML ist eine „Metasprache", das heißt mit SGML werden Regeln zur Verfügung gestellt, um konkrete Auszeichnungssprachen zu definieren. Da SGML einige Eigenschaften hat, die es schwierig machen, SGML

---

15 https://www.w3.org/Style/CSS

16 https://de.wikipedia.org/wiki/JavaScript

17 AJAX = *Asynchronous JavaScript and XML*, bezeichnet ein Konzept der asynchronen Datenübertragung zwischen einem Browser und dem Server. Dieses ermöglicht es, HTTP-Anfragen durchzuführen, während eine HTML-Seite angezeigt wird, und die Seite zu verändern, ohne sie komplett neu zu laden, vgl. https://de.wikipedia.org/wiki/Ajax_(Programmierung).

zu *parsen*[18], also ein Dokument auf seine Semantik hin zu interpretieren und überprüfen, wurde 1998 zur Vereinfachung die eXtensible Markup Language (XML)[19] entworfen. XML besticht als Metasprache einerseits durch eine außerordentlich einfache Syntax,[20] andererseits stehen mit den weiteren Mitgliedern der X-Familie eine ganze Reihe von Tools zur Verfügung, welche XML zu einem äußerst mächtigen Tool in der Strukturierung, Verknüpfung und Übermittlung von Inhalten macht. XML wird deshalb neben der Speicherung strukturierter Daten vorzugsweise für den plattform- und implementationsunabhängigen Austausch von Daten zwischen Computersystemen eingesetzt.

XML ist Teil einer ganzen X-Familie, also Funktionalitäten und Skriptsprachen, die zur Be- und Verarbeitung von XML-Daten genutzt werden können. Aus der X-Familie sind vor allem XPath, XSLT, XSL-FO und XQuery für die weitere Verarbeitung von XML-Dokumenten von Bedeutung. XPath wird für die Navigation in sowie die Selektion von Inhalten aus XML-Dokumenten gebraucht. Mit XSLT können XML-Dokumente in HTML, in ein anderes XML oder in Text transformiert werden. XSL-FO wird für die Vorverarbeitung für ein druckfertiges PDF genutzt und XQuery ist die Abfragesprache für native XML-Datenbanken[21].

---

[18] https://de.wikipedia.org/wiki/Parser

[19] XML 1.0 ist derzeit in der 5.Version gültig, http://www.w3.org/TR/2008/REC-xml-20081126.

[20] Man braucht im Prinzip nur zwei Regeln zu kennen: 1. Alles ist begrenzt. (Elemente werden durch *tags*, *tags* durch spitze Klammern, Attribute durch Anführungszeichen begrenzt.) 2. Alles ist geschachtelt. (Jedes Element muss vollständig in einem anderen Element enthalten sein, es kann keine Überlappung von Elementen geben. Folgerichtig kann es nur ein äußerstes Element, das Wurzel- oder Root-Element, geben.) Darüber hinaus gilt es noch zu beachten, dass Elementnamen zwar frei wählbar sind, aber deren Namen nur aus bestimmten Zeichen bestehen dürfen; dass die Elementnamen case-sensitive sind, das heißt dass Groß- und Kleinschreibung beachtet werden muss; und dass ein Element zwar beliebig viele Attribute haben darf, aber jedes Attribut pro Element nur einmal vorkommen darf.

[21] Native XML-Datenbanken speichern XML unmittelbar. Die in den Geisteswissenschaften am weitesten verbreiteten XML-Datenbanken sind eXist http://exist-db.org, BaseX http://basex.org und Oracle XML DB

Sind die Regeln der XML-Syntax eingehalten, spricht man davon, dass das XML *wohlgeformt* ist. *Wohlgeformtes* XML wird von *validem* XML unterschieden. Eine XML-Datei ist dann valide, wenn nicht nur die Regeln der XML-Syntax, sondern auch die grammatikalischen Regeln einer Anwendungssprache eingehalten sind. Die Grammatik einer Anwendungssprache wird in einer sogenannten *Schemasprache*[22] festgelegt. Die Schemasprache dient dabei zur syntaktischen Beschreibung der Sprache, unter anderem welche Elemente in der Sprache definiert sind, wo und wie häufig diese Elemente vorkommen dürfen, welches Inhaltsmodell die Elemente haben, welche Attribute diese Elemente haben können oder welche Datentypen die Attributwerte haben. Die wichtigsten Schemasprachen sind *XML Schema*[23], *RelaxNG*[24] und *Schematron*[25].

Grundsätzlich wird zwischen *Dokument*-zentriertem und *Daten*-zentriertem XML unterschieden. Das Dokument-zentrierte XML orientiert sich an den Strukturen eines Fließtextes, wie es in Texten und (natürlichsprachigen) Dokumenten vorkommt. Wesentliche Eigenschaft des Dokument-zentrierten XML ist der sogenannte *mixed content*, dem Vorkommen von Elementen, die sowohl Text als auch Kindelemente enthalten. Bei Daten-zentriertem XML enthalten die Elemente entweder Kindelemente oder Text. Daten-zentriertes XML wird als Austauschformat und für zahlrei-

http://www.oracle.com/technetwork/database/database-technologies/xmldb/overview/index.html.

[22] Siehe https://de.wikipedia.org/wiki/Schemasprache_(XML). Eine ältere Variante ist die Dokumenttypdefinition (DTD), https://de.wikipedia.org/wiki/Dokumenttypdefinition. Aufgrund der gegenüber Schemasprachen eingeschränkten Leistungsfähigkeit der DTD werden DTDs im Zusammenhang mit komplexeren XML-Dokumenten allerdings kaum noch verwendet.

[23] Ezell, David; Sperberg-McQueen, C. M.; Thompson, Henry. (28. Oktober 2004). *XML Schema*. World Wide Web Consortium, https://www.w3.org/TR/xmlschema-0.

[24] Clark, James; Makoto, Murata. (3. Dezember 2001). RELAX NG Specification, OASIS, https://www.oasis-open.org/committees/relax-ng/spec-20011203.html.

[25] Jelliffe, Rick. Academia Sinica Computing Centre's Schematron Home Page. Academia Sinica Computing Centre, 2001; Siehe auch http://xml.ascc.net/resource/schematron/schematron.html.

che nicht-textbasierte Datenformate genutzt. Dazu zählen Grafikformate wie *SVG*[26], Geodatenformate wie *Geography Markup Language (GML)*[27] oder *OpenStreetMap (OSM)*[28], Audiodaten wie MusicXML, ein Musik-Notationssystem, Multimediadaten wie MPEG-7-Metadaten und vor allem eine Notationsform für die Daten des semantischen Webs, RDF.[29]

Weitere Formate zur Textauszeichnung, die allerdings eher auf die Darstellung von Dokumenten zielen, sind *LaTeX*[30] und *RTF*[31]. In beiden Formaten können semantische Eigenschaften von Texte durch Markup ausgezeichnet werden, Ziel der Auszeichnung ist aber im Wesentlichen die einheitliche Darstellung von Textteilen mit gleicher Funktion, zum Beispiel von Überschriften, Listen, Fußnoten etc. RTF wird normalerweise mit *WYSIWYG*-Editor[32] eingegeben und ist nicht auf komplexe Layoutgestaltung ausgerichtet, sondern bedient vor allem allgemeine Grundbedürfnisse. LaTeX dagegen adressiert die Anforderungen vor allem wissenschaftlicher Communities an eine Textsatzumgebung. Für wissenschaftliche Publikationen aus den Natur- und Technikwissen-

[26] Scalable Vector Graphics, vgl. Abschnitt zu Bild-Daten.

[27] http://www.opengeospatial.org/standards/gml

[28] https://www.openstreetmap.org

[29] Resource Description Framework, https://www.w3.org/RDF. Mit RDF können logische Aussagen über Ressourcen formuliert werden. In der Regel handelt es sich, wegen der Nutzung im Zusammenhang mit Ressourcen im Internet, um Beziehungen zwischen den Ressourcen. Im RDF wird die Beziehung in Tripeln von Subjekt-Prädikat-Objekt ausgedrückt. Die Kodierung von RDF in XML ist nur eine von mehreren Möglichkeiten, sehr verbreitet sind auch die Einbettung in den HTML-Header sowie in speziellen Datenbanken, sogenannten Triplestores.

[30] https://www.latex-project.org

[31] Rich Text Format, aktuelle Version 1.9.1, http://www.microsoft.com/en-us/download/details.aspx?id=10725. Das von Microsoft 1987 eingeführte Dateiformat für Texte ist vor allem als Austauschformat zwischen Textverarbeitungssystemen gedacht. Es ist im Prinzip durch die moderneren Formate wie das Office Open XML Format (s. das Folgende) überholt, wird aber in allen Textverarbeitungen immer noch als mögliches Speicherformat angeboten.

[32] Die Abkürzung steht für What You See Is What You Get.

schaften, aber auch zum Beispiel für Texteditionen, gibt es zahlreiche Spezialkomponenten, um Formelsatz, mehrfache Apparate, Musiknotensatz usw. zu ermöglichen.

Reine Textdateien sowie Textdateien mit Auszeichnungen sind aufgrund der Plattformunabhängigkeit und Menschenlesbarkeit sehr gut für die *Langzeitarchivierung* geeignet. Weniger gut ist es um die Langzeitarchivierbarkeit der anderen Text-basierten Formate gestellt. Diese beruhen entweder auf proprietären Formaten, die womöglich in Zukunft nicht mehr lesbar sein werden oder auf binären Formaten, die für bestimmte Verarbeitungsszenarien gedacht sind.

Komplexe und binäre Formate für Textdateien

Komplexe Formate liegen beispielsweise in dem E-Book-Format epub und den neueren Formaten der Office-Pakete vor. Die älteren Formate der gängigen Textverarbeitungssysteme der Office-Pakete sowie das als Druckvorstufe genutzte PDF werden als binäre Dateien abgespeichert.

Die komplexen Formate der Textverarbeitungen wie auch das E-Book-Format epub sind im Prinzip Container für XML- und andere Dateien, welche die Textdaten, Stylesheet-Information, gegebenenfalls Bilder und weitere Daten enthalten. Als Standards sind die Formate OpenDocument-Format (ODF) der Open Source Community sowie das Microsoft-Pendant Office Open XML (OOXML) definiert. Microsoft hat dabei das eigene Format 2008 zum Standard erheben lassen, obwohl mit dem Open Source Produkt bereits seit 2005 ein Standard definiert war.[33] Die neueren Microsoft-Dateien erkennt man an der Dateiendung *.docx*, die OpenDocument-Varianten enden auf *.odt*.

[33] Da Konversionen zwischen Formaten immer mit strukturellen oder anderen Verlusten einhergehen, ist es wichtig, die Austauschbarkeit der beiden Formate zu betrachten: In der Regel sind die Open Source Produkte mit sehr funktionalen Konvertern von OOXML-Dokumenten in das eigene Format ausgestattet. Auch können Daten unmittelbar in den Microsoft-Dokumentarten abgespeichert werden. Umgekehrt ist das allerdings nicht der Fall: Auch das neueste Word kann die Open Office Formate weder öffnen noch speichern und dabei jeweils konvertieren.

Während also die komplexeren Formate wiederum Text und Formatangaben im Prinzip getrennt, aber in Container gebündelt speichern, war dies im älteren Microsoft-Word-Format (Dateiendung .doc) nicht der Fall. Dateien dieses Typs sind binär gespeicherte Texte, die nur mit speziellen Editoren geöffnet und angezeigt werden können. Inwieweit die Anzeige dann „korrekt" im Sinne der Microsoft-Implementation ist, hängt von der jeweiligen Software ab. .doc-Dateien dürften immer noch das am weitesten verbreitete Text-basierte Dateiformat sein.

Das aktuell wichtigste binäre Format ist das *Portable Document Format* (PDF). PDF ist als plattformunabhängiges Dateiformat für Dokumente gedacht. Ziel ist die identische Anzeige des Textlayouts auf unterschiedlichen Computersystemen. Daher wird PDF unter anderem im E-Publishing-Bereich häufig verwendet, um zum Beispiel seitengenaue Darstellung zu erreichen und Zitierfähigkeit herzustellen. Die Anordnung der Text- und Bildelemente auf der Seite wird in einer vektorbasierten Beschreibungssprache beschrieben, die frei skalierbar ist. Dadurch wird das Layout unabhängig von der Darstellungsgröße gewahrt. Allerdings kann der Text bei Bedarf, zum Beispiel zur Ansicht auf sehr kleinen Lesegeräten, dennoch umgebrochen werden. In einem PDF ist es möglich, Inhalte als Bild speichern und zusätzlich mit durchsuchbarem Volltext zu hinterlegen. Besonders bei der Repräsentation historischer Quellen kann dies von großem Nutzen sein, da es dadurch möglich wird, das Abbild der Originalquelle mit einem zum Beispiel durch OCR gewonnenen Text zu hinterlegen und für die Volltextsuche zur Verfügung zu stellen. Für alle Betriebssysteme gibt es mittlerweile zahlreiche Programme, mit denen PDFs erzeugt, angezeigt und sogar annotiert werden können. Damit ist PDF zu einem beliebten Arbeitsformat für Dokumente aller Art avanciert. Allerdings sind diese Lesezeichen, Kommentare, Annotationen usw. selbst nicht plattformunabhängig, sondern von der anzeigenden Software abhängig.

Die originalgetreue Wiedergabe erkaufte man anfangs allerdings mit einer Abhängigkeit von der Softwarefirma Adobe. Erst 2005 wurde die Version *PDF/A* speziell für die Anforderungen der *Langzeitarchivierung* als ISO 19005-1:2005 standardisiert und

erst 2008 die Version 1.7 (ISO 32000:1-2008) als ein vollständig offener Standard veröffentlicht. Die meisten Textverarbeitungen und sonstigen Programme, die PDF erzeugen können, stellen PDFs in der Version 1.4, maximal 1.5, her. Die freien Office-Pakete bieten immerhin eine Option, das für die Langzeitsicherung besser geeignete PDF/A-1a zu exportieren. Das größte Manko in der Nutzbarkeit von PDF lag lange Zeit darin, dass man von außerhalb keine Sprungziele in der Datei definieren konnte, obwohl innerhalb der Datei selbst Sprungziele, zum Beispiel Textstrukturen per Inhaltsverzeichnis, erzeugt und angesteuert werden können. Dieser Mangel ist seit 2007 behoben.[34]

Bild-basierte Formate

Bei den Bild-haften Formaten muss man grundlegend zwischen *Raster- und Vektorgrafiken* unterscheiden. Rastergrafiken speichern für jeden Bildpunkt (Pixel) der Grafik eigene Information zu Helligkeit und Farbinformation. Vektorgrafiken hingegen beschreiben die im Bild vorkommenden Formen mathematisch. Die am häufigsten verwendeten Rastergrafikformate sind JPG/JPEG, JPEG2000, TIFF (Tagged Image File Format) und PNG (Portable Network Graphics). Das verbreitetste Vektorbildformat ist SVG (Scalable Vector Graphics).

Rastergrafiken

Die wichtigsten Eigenschaften von Rastergrafiken sind die Bildgröße, Auflösung und die Farbtiefe: Die Bildgröße ergibt sich aus der Anzahl der Bildpunkte in Höhe und Breite. Dagegen ist die Auflösung die Anzahl der Bildpunkte bezogen auf die Größe des abgebildeten Gegenstandes. Die Auflösung wird in der Regel mit der Maßzahl *dpi* (dots per inch) angegeben. Mit dieser Angabe wird angezeigt, wie viele Pixel pro Inch der Vorlage verwendet werden. Für die Digitalisierung historischer Dokumente gilt, dass die Auflösung mindestens 300 dpi, in bestimmten Fällen bis zu

[34] Vgl. https://www.adobe.com/content/dam/acom/en/devnet/acrobat/pdfs/pdf_open_parameters.pdf.

600 dpi oder mehr betragen sollte.[35] Umgerechnet auf eine Vorlage von ungefähr DIN A4 Größe, müssen auf der langen Seite mindestens 3500 Pixel, auf der kurzen Seite mindestens 2480 Pixel gespeichert werden, um den Wert von 300 dpi zu erreichen.[36]

Die sogenannte Farbtiefe bestimmt, wie viele Farben beziehungsweise Helligkeitsstufen in jedem Bildpunkt repräsentiert werden können. Handelt es sich um ein Farbdigitalisat, dann ist die Angabe auf die Tiefe pro Farbkanal bezogen: Gängig ist die Formulierung 8 Bit Farbtiefe, gemeint ist aber 8 Bit je Farbkanal, insgesamt also 24 Bit. Bei Bildern in Graustufen bezeichnet die Farbtiefe die Anzahl von Helligkeitsstufen, die von Weiß bis Schwarz unterschieden werden. Auch hier ist die Tiefe von 8 Bit gängig. Mit diesen 8 Bit können 256 verschiedene Helligkeitsstufen repräsentiert werden.

Ein wichtiger Nebenaspekt der Farbtiefe ist der Farbraum, der von der Anzahl der darstellbaren Farben abgedeckt wird. Der Farbraum bezeichnet die Auswahl der Farben, welche durch die Kodierung definiert sind.[37] Da verschiedene Hard- und Softwarehersteller je eigene Farbräume definiert haben, ist die Feststellung, in welchem Farbraum die eigenen Geräte arbeiten bzw. vorliegende Grafiken gespeichert worden sind, für eine farbgetreue Wiedergabe sehr wichtig. Die Einstellung von Hard- bzw. Software auf einen speziellen Farbraum nennt man Kalibrierung. Bei allen Verarbeitungsschritten von Grafik ist auf die Kalibrierung Wert zu legen, da die menschlichen Sinnesorgane keine objektiven Aussagen über Farbgleichheit oder -ungleichheit erlauben.

---

[35] Vgl. die DFG-Praxisregeln zur Digitalisierung, http://www.dfg.de/formulare/12_151/12_151_de.pdf.

[36] Gerechnet am Beispiel der langen Seite: Die Kantenlänge des abzubildenden Objekt von circa 29,7 cm entspricht circa 11,7 Inch (1 Inch = 2,54 cm). Wenn jedes Inch mit 300 Pixeln repräsentiert werden soll, ergibt sich folgende Rechnung: 29,7 : 2,54 * 300 = 3507,87.

[37] Das Farbmodell, die Art der Farbzusammensetzung, ob additive oder subtraktive Farbmischung, wird in diesem Zusammenhang nicht weiter berücksichtigt.

Für die Qualität eines Bildes sind allerdings nicht nur die Auflösung und Farbtiefe entscheidend, die Möglichkeit der Weiterverarbeitung hängt vor allem an der Frage, ob das Bild verlustbehaftet oder verlustfrei komprimiert wurde. Die Stärke des Formats JPG/JPEG liegt in der geringen Größe der Dateien. Dieses Format wird daher bevorzugt für den Transfer im Internet genutzt. Allerdings erkauft man die geringe Dateigröße in der Regel mit einer verlustreichen Kompression. Bei dieser Kompression wird unter anderem die Farbinformation angrenzender Pixel verglichen. Ist diese relativ ähnlich, wird die Farbinformation für die betroffenen Pixel gemittelt und für diese Pixel nur der eine Mittelwert abgespeichert. Dadurch geht Detailinformation des Ursprungsbildes verloren. Auch bei hoher Auflösung werden durch die angewendeten Algorithmen in der Regel so viele Detail-Informationen zerstört, dass die Bilder über das reine Betrachten hinaus weniger Nutzen haben als unkomprimierte Formate.

JPG/JPEG ist durch die Anwendung von verlustbehafteter Kompression als Archivformat unbrauchbar. Zudem ist die Anfälligkeit von JPG/JPEG gegenüber mechanischen Defekten wesentlich stärker als bei anderen Formaten. Fällt in einem stark komprimierten JPG/JPEG auch nur ein Bit aus, ist ab diesem Bit der ganze Rest des Bildes unbrauchbar und wird ohne Farbinformation als schwarze Fläche dargestellt. Bei unkomprimierten Formaten wäre nur ein einziger Pixel gestört. Erst das neuere Format JPEG2000 kann hohe Kompressionsraten und dadurch geringe Dateigrößen auch durch verlustfreie Algorithmen erreichen. Allerdings sind nicht alle Teile des Formats lizenzfrei. Daher gibt es immer noch zu wenig Standardsoftware, die dieses Format erzeugen oder weiter verarbeiten kann. Aus diesem Grund ist für die Langzeitarchivierung angeraten, Bilder in unkomprimiertem TIFF abzuspeichern. TIFF kann besonders gut mit hohen Farbtiefen und umfangreichen Metadaten umgehen, ist dadurch allerdings in manchen Zusammenhängen überkomplex. Die sehr großen Dateien sind nicht für den massenhaften Austausch über das Internet geeignet. Das Format PNG schließlich stellt eine gute Mischung aus verlustfreier Kompression, Lizenzfreiheit und weiterer nützlicher Eigenschaften dar, weshalb es sich nach JPG/JPEG

zum beliebtesten Bildformat im Internet entwickelt hat. Neuere Browser unterstützen die Anzeige von PNG.

Vektorgrafiken

In Vektorgrafiken werden die abzubildenden Inhalte aus geometrischen Grundformen zusammengesetzt dargestellt. Da für ein kreisförmiges Objekt lediglich die Lage des Kreismittelpunktes, des Radius sowie Informationen über Farbe des Kreises und dessen Füllung gespeichert werden müssen, lassen sich Vektorgrafiken meist deutlich platzsparender speichern. Darüber hinaus sind Vektorgrafiken verlustfrei skalierbar und frei transformierbar. Die Speicherung der Objekte kann durch eine Auszeichnungssprache wie SVG realisiert werden. Bei einer SVG-Datei handelt es sich um eine XML-Datei, welche alle Details zu den enthaltenen Formen, Farben und Füllungen aufnimmt. Da das Bild allerdings erst in der Darstellung entsteht, müssen Anzeigegeräte entsprechende, unvorhersehbare Rechenleistung erbringen.

Typische Anwendungsszenarien für Vektorgrafiken sind 3D- und *CAD-Modellierungswerkzeuge, Schriftgestaltung und Geoinformationssysteme*. Auch audiovisuelle Daten können in einem Vektorgrafik-Format abgelegt werden, dem SWF (Shockwave Flash). SWF ist ein proprietäres Format der Firma Adobe, welches einzelne Inhalte als Vektorgrafik speichern, aber auch Rastergrafik- und Videoformate einbetten und in komprimierter Form speichern kann. Zur Ansicht von SWF-Daten braucht man deshalb ein spezielles Plugin für Browser.

2.2 Datenformate

Von den Dateiformaten sind die Datenformate zu unterscheiden. Während die Dateiformate festlegen, welche Art von Daten und wie Daten technisch gespeichert werden, bestimmen die Datenformate, welche inhaltliche Struktur die Daten haben. Datenformate sind vor allem für Text-orientierte Formate bedeutsam. In ihnen werden die Texte durch Auszeichnungssprachen mit Semantik aufgeladen bzw. die den Texten immanente Semantik wird durch Textauszeichnung explizit gemacht. Diesem Vorgang liegt immer eine Interpretation der Quelle zugrunde. Textauszeich-

nung ist ohne Interpretation und ohne (editorische) Entscheidungen nicht möglich. Schon die Wahl der Auszeichnungssprache legt den Bearbeiter auf bestimmte Sichten auf das Ausgangsmaterial fest. Innerhalb der gewählten Auszeichnungssprache mag es weitere Entscheidungsmöglichkeiten geben, doch der technologische wie der theoretische Rahmen zur Repräsentation der Quellen ist mit der Auswahl festgelegt.

Text Encoding Initiative (TEI)

Die TEI steht sowohl für eine Organisationsform (Text Encoding *Initiative*) als auch synonym für die „Guidelines", die Richtlinien, nach denen Texte mit Elementen der TEI ausgezeichnet werden können (Guidelines for Electronic Text Encoding and Interchange). Die TEI ist das für die (geistes-)wissenschaftliche Nutzung von Texten wohl am weitesten verbreitete und wichtigste Datenformat. Die TEI ist der De-facto-Standard für die Kodierung von Volltexten historischer Dokumente, zum Beispiel für die Langzeitsicherung OCR-generierter Texte *historischer Drucke*[38] oder digitale Editionen. TEI ist zudem das zentrale Austauschformat für Handschriftenbeschreibungen in Deutschland.[39] Die aktuelle Version der Richtlinien wird als P5 (Proposal 5) bezeichnet. TEI-Daten werden in XML kodiert. Die in den Richtlinien beschriebenen Datenstrukturen werden von der TEI in der Metasprache ODD (One Document Does it all) definiert, aus der formale Schemata generiert werden können. Das System der TEI ist modular aufgebaut und unterscheidet Module, Modell- und Attributklassen. Die Module entsprechen im Wesentlichen den inhaltlichen Abschnitten der TEI und ordnen alle Elemente definierten Inhaltsbereichen zu. So werden beispielsweise Elemente, die im Header vorkommen können, von denen unterschieden, die für

[38] So zum Beispiel im Deutschen Textarchiv (DTA), http://www.deutschestextarchiv.de und im Oxford Text Archive (OTA), http://ota.ox.ac.uk.

[39] Für weitere Dokumenttypen gibt es Datenformate, die sich eng an die TEI anlehnen bzw. Teile der TEI-Systematik übernommen haben. So gibt es beispielsweise für Urkunden das Format *Charters Encoding Intitiative* (CEI) oder für Musikeditionen das Format *Music Encoding Initiative* (MEI).

alle TEI-Dokumente nützlich oder aber für bestimmte Dokumenttypen wie Wörterbücher, Handschriftenbeschreibungen oder Transkriptionen vorgesehen sind. Die Modell- und Attributklassen gruppieren Elemente entsprechend ihrer möglichen Position im Text: Als Beispiele sollen hier Bibliographie-artige, Maßzahlen-artige, Namens-artige oder Zitat-artige Klassen für Elemente oder datierbar, typisierbar oder global verfügbar für Attribute ausreichen. Attributwerte können festen Datentypen zugeordnet sein.

Im Format TEI können sowohl Daten als auch Metadaten kodiert werden. Manche Daten können – je nach Kontext – sowohl als Metadaten als auch als Daten fungieren. So kann eine Handschriftenbeschreibung Metadatum sein zu einer Handschrift, beispielsweise im Rahmen der Quellenbeschreibung einer kritischen Edition. Die gleiche Handschriftenbeschreibung kann zugleich Teil der Daten eines Handschriftenkatalogs als Publikation sein. Gleiches gilt für Teile des Textes, der in einer Handschriftenbeschreibung verwendet wird: Der Titel eines Textes kann ein Zitat aus der Handschrift sein und in dieser Funktion im Rahmen der Beschreibung als Rubrik klassifiziert und mit dem TEI-Element <*rubric*> ausgezeichnet sein. Das gleiche Zitat als Teil der Transkription oder Edition des Textes würde aber als Überschrift <*head*> oder in seinen visuellen Eigenschaften als rubriziert oder anderweitig von dem umgebenden Text hervorgehoben mit dem Element <*hi*> ausgezeichnet werden.

Als weiteres Beispiel der Festlegung möglicher Sichtweisen auf Text können die unterschiedlichen Repräsentationsarten von Text in der TEI herangezogen werden. Die TEI ist ursprünglich für die Kodierung von Texten als Werk im Sinn des Konzeptes der *Functional Requirements for Bibliographic Records* (FRBR)[40] konzipiert worden. Ein Text der in dieser Weise nach den Richtlinien der TEI ausgezeichnet wird, folgt in seiner Kodierung grundsätzlich dem

[40] Das FRBR-Konzept unterscheidet Werk (*work*, der Text an sich, unter Umständen als abstrakte Idee eines Textes), Expression (reale Version des idealen Textes, zum Beispiel in Übersetzung), Manifestation (Ausgabe des Textes, zum Beispiel als bestimmte Druckauflage) und Exemplar (*item*, konkretes Buch im Regal einer Sammlung).

Modell eines hierarchisch gegliederten, im Prinzip linearen Zeichenstroms.[41] Erst mit der Einführung der Version P5 hat sich dies geändert. Zunächst wurde das Text-tragende Objekt in den Blickpunkt der Textauszeichnung gerückt, danach auch noch eine zeitliche Abfolge, in der der Text auf der Schreiboberfläche aufgebracht worden ist. Diese Sichten auf das Dokument werden unter anderem durch unterschiedliche Top-Level-Elemente repräsentiert.

Grundsätzlich besteht eine TEI-Datei aus dem TEI-Header und einem oder mehreren weiteren, die Quelle repräsentierenden Elementen. Der TEI-Header (Element <*teiHeader*>) enthält die bibliographischen Angaben zu dem elektronischen Dokument und den zugrunde liegenden Quellen. Die weiteren Top-Level-Elemente entsprechen den oben genannten Sichten der TEI auf den Text: Das Element <*text*> steht für den Text als Werk und nimmt die Transkription, Edition oder andere Volltextrepräsentation der Quelle auf; das Element <*facsimile*> repräsentiert die Quelle als Folge von Bildern bzw. Oberflächen und bildet damit den Text als Objekt im Bild ab; das Element <*sourceDoc*> verknüpft Bild und Transkription/Edition miteinander und erlaubt eine zeitliche Ordnung des Aufbringens des Textes zu repräsentieren.

Die zentrale Frage bei der Interpretation und (Nach-)Nutzung von TEI-Dokumenten ist deren Konformität. Einerseits gibt es das Problem, dass die TEI für viele Phänomene mehr als eine Kodierungsmöglichkeit mit gleichem oder zumindest sehr ähnlichem semantischen Gehalt erlaubt und daher Kodierungen sehr unterschiedlich ausfallen können.[42] Welches der Elemente genutzt wird kann vom Kontext abhängen oder von lokalen Konventionen. Für

[41] Die theoretische Grundlage ist das OHCO-Modell, das Modell der „Ordered Hierarchy of Content Objects“. Dieses Modell findet seine technische Übereinstimmung in der Anwendung von XML. Zu einer grundsätzlichen Kritik des Modells vgl. Allan Renear (1993).

[42] Die TEI bietet in solchen Fällen generische und alternativ speziellere Elemente an. Zur Kodierung von Personennamen stehen beispielsweise das spezielle Element persName, das etwas allgemeinere name und das sehr generische rs (referencing string) zur Verfügung. Die letzteren beiden könnten über ein Attribut „type“ genauer spezifiziert werden.

eine einheitliche Verarbeitung und Ausgabe, aber auch für die konsistente Verwendung in größeren Projekten ist es daher wichtig, die Entscheidungen über die Verwendung von Elementen gründlich zu dokumentieren. Andererseits schafft das System der TEI mit seinem modularen Aufbau die Voraussetzung, die TEI kontextabhängig zu nutzen und beispielsweise nur eine Auswahl an Modulen und Klassen in einem eigenen Schema zusammen zu fassen und zu nutzen. In der TEI spricht man dann von „Customisation". Dies kann nun dazu führen, dass Dokumente, obwohl sie die gleiche Art von Quellen repräsentieren, dennoch unterschiedliche Strukturen aufweisen. Hierunter würde die Interoperabilität der Daten leiden. Wichtig ist daher für eine solche Anpassung, dass gegenüber dem allgemeinsten Schema valide Daten erzeugt werden.

Encoded Archival Description (EAD)

Ein weiteres wichtiges Datenformat im Kontext historischer Forschungen ist *Encoded Archival Description* (EAD). EAD wurde für die Kodierung von archivischen Findmitteln entwickelt. Auch EAD wird in XML kodiert. Der Volltext der Findmittel bildet die hierarchische Topik ab, in der Archive traditionell organisiert sind. Daraus folgt, dass in EAD eher die Struktur von Sammlungen als das einzelne Dokument beschrieben wird.

EAD ist zum Beispiel das zentrale Austauschformat der Datenbank *Kalliope*[43]. EAD ist als Format eng verknüpft mit dem Format Encoded Archival Context (EAC). EAC ist ein Standard zur Kodierung von Meta-Informationen über die Herkunft und Benutzungsgeschichte von Archivgut.

Weitere Datenformate

Neben TEI und EAD gibt es noch weitere Datenformate, die in spezifischen musealen und bibliothekarischen Kontexten eine Rolle spielen. Aufgrund ihrer Spezifität werden sie hier nicht ausführlicher vorgestellt. Zu nennen wären zum Beispiel: *Lightweight*

[43] http://kalliope.staatsbibliothek-berlin.de

Information Describing Objects (LIDO)[44], *CIDOC Conceptual Reference Model (CIDOC-CRM)*[45] oder *Metadata Encoding & Transmission Standards (METS)*[46].

2.3 Herausforderungen: Das Semantic Web

Durch die zunehmende Verfügbarkeit großer Datenmengen steigt die Notwendigkeit, diese Daten automatisiert miteinander in Verbindung treten zu lassen, da das manuelle Verknüpfen immer unrealistischer wird. Die semantische Aufladung von Daten, mit der diese Funktionalität ermöglicht werden soll bzw. die Auswertung dieser Aufladung wird als „*Semantic Web*" bezeichnet. Die oben besprochene Textauszeichnung fügt Texten bereits strukturelle Semantik hinzu, so dass beispielsweise Autorennamen von Werktiteln unterschieden werden können. Welche Person allerdings mit der Nennung als AutorIn tatsächlich gemeint ist, mag damit aber immer noch unklar bleiben. Um eine Person, aber auch Institutionen, Orte und sogar Bildinhalte oder fachspezifische Begriffe eineindeutig zu identifizieren, bedarf es der Verknüpfung dieser Entitäten mit Normdaten.

Für die meisten Entitäten sollte in Deutschland die *Gemeinsame Normdatei*[47] (GND) genutzt werden. Die GND wird an der Deutschen Nationalbibliothek gepflegt und kann über den OPAC der DNB abgefragt werden. Jedes Suchergebnis wird in der rechten Spalte von Facetten begleitet, die das weitere Eingrenzen des Suchergebnisses ermöglichen. In der rechten Spalte findet sich unter anderem auch der Eintrag „Alle Normdaten". Folgt man den darunter gelisteten Links, gelangt man zu den Normdaten. Die Normdaten werden zunächst als normale Webseite angezeigt, doch sind weitere, maschinenlesbare Versionen verfügbar. Jeder Datensatz kann automatisiert im MARC21-XML- oder RDF-Format ausgegeben werden. Da die GND durch Zusammenlegen

[44] http://network.icom.museum/cidoc/working-groups/lido

[45] http://www.cidoc-crm.org

[46] http://www.loc.gov/standards/mets

[47] http://www.dnb.de/DE/Standardisierung/GND/gnd_node.html

mehrerer Vorgängernormdateien entstanden ist, besteht an den Daten immer noch großer Harmonisierungs- bzw. Bereinigungsbedarf.

Die Normierungsanstrengungen, welche die DNB auf nationaler Ebene betreibt, spiegeln sich im internationalen Umfeld im Projekt *Virtual International Authority File*[48] (VIAF). In VIAF werden die Normdateien aller beteiligten Länder zusammengespielt und möglichst aufeinander gemappt. So soll es möglich werden, landes- bzw. sprachentypische Bezeichnungen für Personen und andere Entitäten automatisiert abzugleichen und abzufragen.

Im Bereich geographischer Entitäten sind die derzeit maßgeblichen Normdatensysteme *GeoNames*[49] sowie der Getty *Thesaurus of Geographical Names*[50] (TGN). In beiden Systemen ist es möglich, auch historische geographische Bezeichnungen zu recherchieren und die Entitäten mit Geokoordinaten zu verknüpfen. Diese können dann genutzt werden, um Daten in einem geographischen Informationssystem, beispielsweise einer Karte oder einem Geobrowser anzeigen zu lassen.

3. Fazit

Für die wissenschaftliche Beschäftigung mit Texten sind Texte in reinen Textformaten mit Textauszeichnung am besten geeignet. Die explizite Zuweisung von Semantik erlaubt Möglichkeiten der Verarbeitung und Suche, die über die Anzeige, den Druck und die Volltextsuche weit hinausgehen. XML, dessen Trennung von Struktur, Inhalt und Form und die Familie an Tools aus der X-Familie macht es zum derzeit bevorzugten Format. Auch für die Langzeitarchivierung sind reine Textformate die beste Option.

Da XML eine Metasprache ist, in der konkrete Auszeichnungssprachen definiert werden können, muss für Dokumenttypen eine angemessene Auszeichnungssprache ausgewählt werden. Die TEI ist eine solche Sprache und der de-facto Standard für die Kodie-

[48] http://www.viaf.org

[49] http://www.geonames.org

[50] http://www.getty.edu/research/tools/vocabularies/tgn/index.html

rung von (geisteswissenschaftlichen) Volltexten wie beispielsweise digitale Editionen.

Für fortgeschrittene Publikationsanforderungen wie mehrfache Apparate ist LaTeX eine weitere Auszeichnungssprache, die bisher vor allem in den Natur- und Technikwissenschaft eingesetzt wird. Aber auch für geisteswissenschaftliche Anforderungen gibt es zahlreiche Spezialkomponenten. Als Druckvorbereitungsstufe und für die Onlinepublikation ist in der XML- sowie LaTeX-Verarbeitung PDF als Format am weitesten verbreitet. PDF ist zwar grundsätzlich ein proprietäres Format, mit PDF/A existiert aber auch eine standardisierte und für die Langzeitarchivierung vorgesehene Version.

Auch bei Bild-haften Formaten gibt es wichtige Unterschiede zwischen Pixel-basierten und Vektorgrafiken sowie zwischen Publikations- und Archivformaten. Das JPG-Format ist wegen der geringen Dateigröße für die Übertragung von Bildern über das Netz beliebt. Die geringe Dateigröße wird in der Regel allerdings durch eine verlustbehaftete Komprimierung erzielt, welche Informationen des Originalbildes vernichtet und die wissenschaftliche Untersuchung bzw. Bearbeitung der Bilder beeinträchtigt. Daher werden das Format TIFF oder verlustfrei komprimierte Bilder im Format JPG2000 für die Langzeitarchivierung empfohlen. Für die Darstellung im Web erfreut sich das ebenfalls verlustfrei komprimierbare Format PNG immer größerer Beliebtheit. Alle diese Formate sind Pixel-basiert. Das Format SVG ist das am weitesten verbreitete Vektorgrafikformat. Vektorgrafiken lassen sich verlustfrei komprimieren und beliebig skalieren. Allerdings sind Vektor-basierte Formate nicht geeignet, um Fotografien oder Scans historischer oder anderer Dokumente zu speichern.

Die Auflösung von Digitalisaten historischer Dokumente muss für eine sinnvolle wissenschaftliche Nutzung mindestens 300 dpi betragen.

Literaturhinweise

Born, Günter, Referenzhandbuch Dateiformate. Grafik, Text, Datenbanken, Tabellenkalkulation. 3. Auflage, Bonn u.a. 1995.

Charters Encoding Intitiative (CEI), http://www.cei.lmu.de.

Encoded Archival Description (EAD), http://www.loc.gov/ead.

Functional Requirements for Bibliographic Records. IFLA Study Group on the FRBR (IFLA Series on Bibliographic Control 19), München 1998, https://www.ifla.org/files/assets/cataloguing/frbr/frbr_2008.pdf.

infokit: Digital file formats, https://www.jisc.ac.uk/website/legacy/digital-media.

infokit: Colour Management for Digitisation Projects, http://www.jiscdigitalmedia.ac.uk/infokit/colour-management/colour-management-for-digitisation-projects-home.

infokit: Metadata, https://www.jisc.ac.uk/guides/metadata.

Music Encoding Initiative, http://music-encoding.org.

Projekt Succeed: Recommendations for metadata and data formats for online availability and long-term preservation, Version 1.1, 2014, http://www.succeed-project.eu/sites/default/files/deliverables/Succeed_600555_WP4_D4.1_RecommendationsOnFormatsAndStandards_v1.1.pdf.

Renear, Alan; Elli Mylonas, David Durand, Refining our Notion of What Text Really Is: The Problem of Overlapping Hierarchies, http://cds.library.brown.edu/resources/stg/monographs/ohco.html.

Roeder, Torsten, Alpha into Alif. Schnittstellen zwischen Schriftkunde und Informatik am Beispiel von Unicode im Glossarium Graeco-Arabicum, in: Studia graeco-arabica 5 (2015) S. 345–363, https://www.academia.edu/14639413.

TEI P5: Guidelines for Electronic Text Encoding and Interchange. The TEI Consortium (ed.) 2007–2015, http://www.tei-c.org/Guidelines/P5.

Torsten Schaßan ist wissenschaftlicher Mitarbeiter der Abteilung Handschriften und Sondersammlungen der Herzog August Bibliothek Wolfenbüttel. Zudem ist er Gründungsmitglied des Instituts für Dokumentologie und Editorik (IDE).

Zitation: Torsten Schaßan, Digitale Quellen – Datei- und Datenformate, in: Clio Guide – Ein Handbuch zu digitalen Ressourcen für die Geschichtswissenschaften, Hrsg. von Laura Busse, Wilfried Enderle, Rüdiger Hohls, Thomas Meyer, Jens Prellwitz, Annette Schuhmann, 2. erw. und aktualisierte Aufl., Berlin 2018 (=Historisches Forum, Bd. 23), S. A.6-1 – A.6-26, DOI: 10.18452/19244.