Tools und Methoden der Formaterkennung aus Sicht der digitalen Langzeitarchivierung

Rolf Lang

Landesarchiv Baden-Württemberg
Arsenalplatz 3
71638 Ludwigsburg
rolf.lang@la-bw.de

MS-DOS und Windows-Anwender sind es gewohnt, dass der Name einer Datei gleichzeitig ihren ‚Typ' anzeigt. Auf Unix-Systemen konnte sich diese Konvention nur teilweise durchsetzen. In vielen Fällen zählt allein der Inhalt einer Datei, getreu dem von Goethe formulierten Motto: „Name ist Schall und Rauch".

1. Grundproblem

Digitale Archive haben die Aufgabe, Daten über eine sehr lange Zeit (> 100 Jahre) zu bringen. Daten alleine zu konservieren reicht allerdings nicht, da auch eine Nutzung möglich sein muss. Diskutiert werden im Wesentlichen zwei Erhaltungsstrategien: Migration und Emulation. In der Praxis greifen digitale Archive zumeist auf die Migrationsstrategie zurück, d.h. zu dem Zeitpunkt, an dem ein Format von der dann aktuellen Software nicht mehr verarbeitet werden kann, wird es in ein neues, möglichst standardisiertes, nicht proprietäres und damit langlebiges Format überführt. In diesem Beitrag wird der Weg der Formatmigration und der notwendigen Voraussetzungen näher betrachtet:

Für eine Migration ist es wichtig, zunächst die Ausgangsdatei zu erkennen. Erkennen bedeutet, von welcher Anwender-Software und mit welcher Version diese Datei erstellt wurde. Kommt eine Anwender-Software in die Jahre und wird nicht mehr unterstützt bzw. ist nicht mehr auf dem verwendeten Betriebssystemen lauffähig, so ist es höchste Zeit zu handeln.

Schritt 1 ist die Identifizierung einer alternativen Anwender-Software und Schritt 2 die Überführung des Ausgangsformates in das neue Zielformat. Natürlich darf es zu keinem Informationsverlust führen, wenn die neue Software das konvertierte Format nutzt. Was heißt nun Informationsverlust?

Ist ein Informationsverlust gegeben, wenn beispielsweise bei der Migration eines MSOffice Dokumentes die Metadaten von MsWord nicht nach Adobe PDF/A gelangen?

Die Frage kann unterschiedlich beantwortet werden. Also benötigt man einerseits eine klare Vorstellung dessen, was wichtig und erhaltenswert ist, und andererseits die Kenntnis, was davon tatsächlich verwendet wurde.

Um nun sicherzustellen, dass die Formatmigration korrekt erfolgte gibt es folgende Ansätze:

A) Man zertifiziert ein Migrationstool auf seine Tauglichkeit.

In diesem Fall wird das Migrationstool validiert auf korrekte Überführung. Alle erforderlichen Eigenschaften des Ausgangsformates werden überprüft und gewichtet bewertet, ob sie ins Zielformat überführt werden konnten. Danach gilt dieses Tool als geeignet für die Überführung aller Dateien für das gewählte Format.

B) Man kontrolliert den Prozess.

Schon zum Zeitpunkt der Archivierung der Quelldatei wird festgehalten, welche wichtigen Eigenschaften für ein Format erhaltenswert sind und welche vorkommen.Nach der Migration können diese signifikanten Eigenschaften erneut erfasst werden und mit den Ausgangsdaten verglichen werden, um das Resultat des Migrationsprozesses zu prüfen und zu dokumentieren.Im folgenden sollen nun übliche Verfahren näher betrachtet und verglichen werden. Beschrieben wird die Funktionsweise der Formaterkennung und wie die Formateigenschaften ermittelt werden.

2. Methoden

File extensions

Diese sind vor allem in der Windows Umgebung gebräuchlich. Seit dem ersten Erscheinen von MS-DOS im Jahre 1981 gibt es Extensions, damals in der 8.3 Notation. 8 Zeichen für den Dateinamen und 3 Zeichen für die Erweitererung. Auch heute sind noch die meisten `file extensions' nur nur 3 Zeichen lang, obwohl heutige Dateisysteme auch mehr Zeichen erlauben. Es ist nicht verwunderlich, dass manche Extension wie „doc“ von verschiedenen Applikationen sowohl gleichzeitig als auch zeitlich versetzt genutzt werden.

Daraus folgt: Aus der Extension kann nur ins seltenen Fällen auf den Inhalt der Datei gefolgert werden. Wenn es also nicht gelingt mit der Auszeichnung über die Extension hilft nur eine nachträgliche Analyse des Inhalts.

Magic Numbers

Hierbei werden Dateiformate erkannt auf Basis einer Mustererkennung. Für eine Liste an Programmen sind signifikante Muster definiert. Beispielsweise beginnt jede PDF Datei mit der Zeichenfolge %PDF-. Die Startposition für die Mustererkennung ist frei definiert, ebenfalls das Auffinden für sekundäre Informationen wie beispielsweise die Formatversion.

Obwohl dieses Verfahren keine exakte Formatidentifizierung sicherstellt, ist die Trefferquote doch erstaunlich hoch und findet Einsatz in diversen Tools.

3. Tools

Unix Systeme: file

Bei der Identifizierung unbekannter Datenformate hilft das Programm file. Ist der übergebene Dateiname eine ordentliche Datei (‚regular files'), wird der Inhalt nach bestimmten Bytefolgen, so genannten Signaturen oder ‚magic numbers' durchsucht.

Passt ein Eintrag aus der Liste, gilt die Identifizierung als erfolgreich.

Einträge in der magic Datei beanspruchen jeweils eine Textzeile. Diese kennt mehrere Felder, jeweils durch einen Tabulator oder 4 Leerzeichen getrennt. Das erste Feld enthält eine Zahl, welche festlegt mit welchem Versatz die ‚magic number' vom Dateianfang auftaucht, typischerweise 0. Das zweite Feld beschreibt unterschiedliche Typen wie Byte, String. Dann folgt im dritten Feld die eigentliche ‚magic number' und im 4. Feld das Dateiformat.

Für PDF sieht der Eintrag in der Datei magic wie folgt aus:

```
0       string          %PDF-           PDF document
>5      byte            x               \b, version %c
>7      byte            x               \b.%c
```

Zeilen, die mit ‚>' beginnen, zeigen sekundäre Einträge, falls das Format schon bestimmt ist. Diese können geschachtelt sein.

Falls diese Mechanismen nicht greifen, folgt ein letzter Versuch: Byte für Byte wird untersucht, ob die Datei Text oder Binärdaten enthält. Falls die Datei nicht druckbare Zeichen enthält, wird sie als ‚data' klassifiziert, was so viel bedeuted wie ‚unidentifizierbar'. Im Falle von Text wird versucht herauszufinden, um welche Art von Text es sich handelt: normales Englisch, Quelltexte in verschiedenen Programmiersprachen und so weiter.

Nahezu jedes kommerzielle unix besitzt seine eigene, unabhängig weiterentwickelte Version von file.

MS Windows: TrID

Ist eine Windows basierte Umsetzung zur Formaterkennung auf Basis von ‚magic numbers'. Derzeit werden 3189 XML Definitionen verwendet zur Identifizierung (Stand 31.03.08).

Es gibt sowohl eine grafische als auch eine Kommandozeilenversion.

Die Beschreibung der ‚magic numbers' erfolgt hier in einer XML Datei. Für PDF gibt es nur einen Eintrag. Diese Beschreibung identifiziert zwar PDF, erkennt aber nicht die PDF-Version.

Apache

Auch ein Webserver muss den Inhalt seiner Dateien kennen, um seinen Clients vor den eigentlichen Daten eine Kennzeichnung zu schicken zu können. Diese Kennzeichnung (mime type) besteht aus einem Haupttyp und einem Untertyp, getrennt durch einen Schrägstrich. (z.B.: text/html, image/gif). Auszeichnung werden gemäß RFC 2045, 2046, 2047, 2048, und 2077 vorgenommen und sind im ‚internet media registry' der IANA registriert.

Apache hat zur Formaterkennung ebenfalls eine freie Version der unix file Erkennung als Grundlage genutzt. Damit werden Formate den mime types zugeordnet. Allerdings ist für einen Webserver vollkommen unerheblich, welche Version das jeweilige Formate besitzt.

JHOVE

Das JHOVE Project ist eine Zusammenarbeit von JSTOR und der Harvard University Library.

Es beherrscht die Identifikation, die Validierung und die Charakterisierung von Dateien mittels folgender Standard Modulen (AIFF, ASCII, BYTESTREAM, GIF, HTML, JPEG, JPEG 2000, PDF, TIFF, UTF-8, WAVE, XML).

Für Microsoft Formate wie MS-Office gibt es keine Module. Intern werden auch ‚magic-numbers' verwendet, um das Format zu identifizieren, allerdings sind diese nicht öffentlich zugänglich. Ebenso fehlt eine Java Dokumentation der verwendeten Methoden im Stile von javaDoc.

Das Projekt gilt als abgeschlossen und eine Neuauflage als JHOVE2 ist geplant. JHOVE ist in der Lage, nicht nur Formate zu identifizieren, sondern auch benutze Eigenschaften daraus zu extrahieren. Dies unterscheidet JHOVE wesentlich von allen anderen Verfahren.

Pronom - DROID

Pronom ist das Format Register von „The national Archives“ und hat derzeit 366 Formate zur Format Identifizierung veröffentlicht. (Stand DROID_SignatureFile_V13 2007-09-26T17:05:59)

Hier ist eine XML Datei die Grundlage für die Formaterkennung. Die Sequenz der ‚magic Numbers’ wird in XML definiert und einer InternalSinatureID zugeordnet. Pronom erkennt detailliert PDFs der Version 1.0 bis 1.6 Weitere 8 Formate für pdf/A und pdf/X sind definiert, werden aber nicht erkannt.

4. Bewertung der Verfahren

Für welches Verfahren hat sich nun das Landesarchiv entschieden? JHOVE ist sehr gut was die Charakterisierung angeht, hat aber folgende Nachteil: Es sind nur wenige module zur Formaterkennung derzeit implementiert. So fehlen zum Beispiel alle Microsoft Office Formate. TrID ist plattformabhängig und daher ungeeignet. Pronom ist gut dokumentiert und integrierbar, unternimmt aber keine Charakterisierung.

Damit blieb die Wahl nur zwischen JHOVE und Pronom. Das Landesarchiv Baden-Württemberg hat sich für eine Eigenentwicklung (IngestList) entschieden, das auf der Basis von Promon und JHOVE eine Formaterkennung durchführt, aber auch weitere Funktionen enthält:

- Erfassung signifikanter Eigenschaften
- Protokollierung der Prozesse
- Validierung der Daten
- Fortschreiben und Vergleich von signifikanten Eigenschaften nach Migration

Das Tool unterstützt zentrale Archivierungsprozesse und hilft, digitale Archivalien glaubhaft über eine sehr lange Zeit zu bringen.

Quellen

iX 1/2002	Zauberhaft Michael Riepe – Identifizierung von Datei- und MIME-Typen mit file
TrID	http://mark0.net/soft-trid-e.html
IANA	http://www.iana.org/assignments/media-types/
file extension	http://ww.file-extensions.org
fileinfo	http://www.fileinfo.net/
Pronom	http://www.nationalarchives.gov.uk/pronom/
JHOVE	http://hul.harvard.edu/JHOVE/