



puter kann sich die Rohdaten herausziehen und ohne Informationsverlust weiterverarbeiten.

Programmierte Informationsverluste

Vielerorts arbeitet das Output-Management jedoch immer noch druckorientiert, das heißt, die Systeme erzeugen primär Druckströme. Es geht also lediglich um die visuelle Repräsentation der Dokumente, die Steuerung der Drucker und eventuell der Verpackungsmaschinen. Die aus dem Druckdatenstrom extrahierten Schriftstücke sind quasi Abfallprodukte, die man in PDF konvertiert und an den Empfänger übermittelt – der elektronische Austausch wird sozusagen nachgerüstet. In diesem Schritt können bereits viele Informationen verloren gehen. Auch wenn ein Mensch auf den ersten Blick sieht, dass es sich beispielsweise um eine Rechnung handelt, ist der Rechner hier zunächst blind.

Er muss daher in den Metadaten der Datei nachsehen, ob im Feld für den Dokumententyp ein aussagekräftiger Wert steht. Oder er extrahiert den gesamten Text und sucht nach der Zeichenkette „Rechnung“. Auch wenn er sie findet, könnte es sich trotzdem um eine Mahnung handeln, die sich auf eine Rechnung bezieht. Manche Anwendungen verankern die einzelnen Seiten sogar als Bitmaps (Bilder) in das PDF, vielleicht um vermeintlich Speicherplatz zu sparen. In diesem Fall kann der Empfänger nicht einmal mehr den Fließtext herausfiltern und muss wie beim Papierdokument zum OCR-Werkzeug greifen. Es entsteht ein Medienbruch, weil trotz elektronischem Transfer zu wenig Metadaten zur Verfügung stehen.

Wer den unstandardisierten Dokumentenaustausch stärker automatisieren will, muss sich mit Adobes Extensible Metadata Platform, kurz XMP, befassen (siehe Kasten „Stan-

XMP: Einbetten von Metadaten in Dokumente

Angeheftet

Jeremias Märki, Jörg Palmer

Ohne aussagekräftige Metadaten lässt sich die Dokumentenverarbeitung nicht durchgängig automatisieren. Das Format XMP spielt hier eine wichtige Rolle, indem es Regeln zum Umgang mit beschreibenden Informationen festlegt.

Unternehmen verarbeiten Rechnungen, Schadensmeldungen, Verträge, Kündigungsbestätigungen und andere Schriftstücke zunehmend digital. Input- und Output-Management müssen infolgedessen zusammenwachsen. Grundlage für ein weitgehend automatisiertes Dokumentenmanagement sind Metadaten, externe Informationen über ein Dokument. Das können die Art oder der Typ sein, das Ausstellungsdatum, der Absender, der Bezug zu anderen Vorgängen und Ähnliches. Ein typisches Szenario im Input-Management:

Die Anwender scannen Dokumente, lesen ihren Inhalt per OCR (Optical Character Recognition) aus und legen sie in einem Archiv oder DMS ab. Gleichzeitig müssen sie mittels Barcodes, Regeln und Heuristiken die zugehörigen Metadaten ermitteln und aufbewahren. Nur so lässt sich der Schriftverkehr korrekt zuordnen, kategorisieren und jederzeit wieder abrufen. Ohne aufwendige manuelle Kontrollen geht es derzeit meist noch nicht.

Es existieren zwei Grundvarianten des Dokumentenaustausches: klassisch in Papierform und digital in einem

XML- oder EDIFACT-Format. Aktuelle Standards wie ZUGFeRD sollen beide Welten vereinen [1]. Das seltsame Akronym steht für „Zentrale User Guidelines des Forums elektronische Rechnung Deutschland“. ZUGFeRD basiert auf PDF/A-3 und schreibt vor, dass die XML-Rohdaten, auf denen das Dokument beruht, als Attachment in die PDF-Datei eingebettet werden (siehe „Alle Links“). Man will so manuelle Eingriffe in die automatische Verarbeitungskette vermeiden. Die Anwender können solch ein Dokument lesen und drucken, der Com-

dards und Normen“). Diese Spezifikation liefert Richtlinien zum Einbetten von Metadaten in Dokumente. Zwar dürften PDFs der häufigste Anwendungsfall sein, aber auch PostScript-, JPG-, PNG-, TIFF-, HTML- und AFP-Dateien lassen sich mit Kontextinformationen anreichern. Ein großer Vorteil von XMP-Paketen, die nichts anderes sind als XML-Dateien: Sie enthalten einen eindeutigen Marker und werden praktisch immer im Klartext abgelegt. So kann ein Programm, auch ohne das Dateiformat zu kennen, die Metadaten auslesen. Wegen der Klartextform sollte man Vorsicht walten lassen und keine potenziell sensiblen Daten wie Versichertennummern hineinschreiben (siehe Listing).

XMP definiert einen Satz universeller Kerneigenschaften wie Titel, Erzeuger, Thema, Datum, Identifikator, Sprache und Beschreibung. Dafür greift die Spezifikation zurück auf

vorhandene Methoden und Standards zur Beschreibung von Metadaten (Ontologien) wie Dublin Core, IPTC und Exif (siehe Kasten „Standards und Normen“). Darüber hinaus kann der Anwender individuelle Merkmale definieren, zum Beispiel Kunden- und Policennummer, Gültigkeit eines Dokuments, Fälligkeit einer Rechnung oder Name/Version einer Dokumentvorlage.

In der Praxis noch nicht verbreitet

Anwendung findet XMP derzeit vor allem bei PDF/A. Der ISO-Standard enthält Hinweise zum Definieren und Hinterlegen von Metadaten. Beispielsweise macht er das XMP-Paket für PDF/A-Dokumente obligatorisch und rät, einen eindeutigen Identifikator zu benutzen. Weiterhin empfiehlt PDF/A, Herkunftsinformationen durch den gesamten Do-

Listing: Beispielhaftes XMP-Paket

```
<?xpacket begin="<U+FEFF>" id="W5M0MPCehiHzeSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:meta:">
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    <rdf:Description xmlns:dc="http://purl.org/dc/elements/1.1/"
      rdf:about="">

    <dc:language>de</dc:language>
    <dc:title>Whitepaper: XMP</dc:title>
    <dc:creator>Compant AG</dc:creator>
    <dc:type>Whitepaper</dc:type>
    <dc:identifier>83db4e23-e840-4b8d-b93f-c3b96cc54ae7</
      dc:identifier>

    <dc:subject>
      <rdf:Bag>
        <rdf:li>XMP</rdf:li>
        <rdf:li>Metadaten</rdf:li>
      </rdf:Bag>
    </dc:subject>
    <dc:date>2013-12-11T15:50:15+02:00</dc:date>
    <rdf:Description>
    <rdf:Description xmlns:pdf="http://ns.adobe.com/pdf/1.3/"
      rdf:about="">

      <pdf:PDFVersion>1.4</pdf:PDFVersion>
    </rdf:Description>
    <rdf:Description xmlns:xmp="http://ns.adobe.com/xap/1.0/"
      rdf:about="">

      <xmp:CreateDate>2013-12-11T15:50:15+02:00</xmp:CreateDate>
      <xmp:MetadataDate>2013-12-11T15:50:15+02:00</xmp:MetadataDate>
    </rdf:Description>
    </rdf:RDF>
  </x:xmpmeta>
<?xpacket end="r"?>
```

kumenten-Workflow mitzuführen. Alle individuellen Eigenschaften muss der Benutzer bei PDF/A-Dateien in einem eingebetteten Schema beschreiben. Dieses Prozedere dürfte ein Grund dafür sein, dass XMP sowie der ausgiebige Einsatz des Standards in der Praxis viel zu kurz kommen.

Dabei ist die Sache weniger kompliziert, als es den Anschein hat. ZUGFeRD setzt hier einen Akzent und ermutigt die Anwender, XMP-Erweiterungsschemata zu verwenden. Die schon angesprochene Verschmelzung von Eingangs- und Ausgangsverarbeitung als Folge des elektronischen Dokumentenaustausches verlangt allerdings einige technische und organisatorische Änderungen. Bisher ging es im Output-Management meist nur darum, die eigenen Belege effizient und zuverlässig zu produzieren und zu senden. Inzwischen muss sich der Versender ebenso Gedanken darüber machen, welche Daten dem Empfänger das Verarbeiten der Dokumente erleichtern können. Denn bei jedem Medienbruch gehen Informationen verloren, die im Nachgang mühsam wiederhergestellt werden müssen.

Schon ein minimales Datenset kann das elektronische Dokumentenmanagement erheblich voranbringen. Ein paar sinnvolle Werte – etwa ein eindeutiger Identifikator für jeden Beleg oder ein Indikator des Dokumententyps (Rech-

nung, Mahnung et cetera) – helfen schon, das Input-Management zu vereinfachen. Allerdings sollte man die verfügbaren Felder nicht für sachfremde Zwecke missbrauchen, was leider oft vorkommt. Der Einsatz von XMP ist jedenfalls ein wichtiger Schritt auf dem Weg zur durchgängigen Automatisierung auf der Eingangs- sowie auf der Ausgangsseite. Natürlich könnte man auch ohne XMP Dokumente mit Metadaten versehen, allerdings nur in abgespeckter Form und minderer Qualität. Für PDF-Dokumente gibt es zu XMP derzeit keine Alternative. (jd)

Jeremias Märki

ist selbstständiger Softwareentwickler und IT-Berater in Luzern (www.jeremias-maerki.ch).

Jörg Palmer

ist Chief Development Officer bei der Compant AG in Böblingen.

Literatur

- [1] Vadim Bauer; Standards; Fakturierungsautomat; ZUGFeRD – ein Datenformat für den elektronischen Rechnungsaustausch; iX 8/2014, S. 98

Alle Links: www.ix.de/ix1411110

Standards und Normen

XMP (Extensible Metadata Platform) ist ein Standard zum Einbetten von Metadaten in Dateien. Adobe hat ihn im Jahr 2001 vorgestellt und erstmals in den Acrobat Reader 5 integriert. Im Februar 2012 wurde der Kern der Spezifikation als ISO-Standard 16684-1 veröffentlicht. Alle Adobe-Produkte sowie verschiedene Anbieter von Redaktionssystemen und anderer Software nutzen XMP. Das Format definiert unter anderem die Sprache des Dokuments, das Erstellungsdatum, den Verfasser und den Namen der ausstellenden Firma.

XMP basiert auf offenen Standards und bettet die vom World Wide Web Consortium (W3C) entwickelte formale Sprache RDF (Resource Description Framework) in Binärdaten ein. Damit lassen sich die Metadaten in verschiedenen Applikationen nach einem einheitlichen Schema so integrieren, dass andere Programme die Dateien weiterhin lesen können. Mit RDF werden Ressourcen (Objekte, Positionen, Personen) sowie ihr

Verhältnis zueinander beschreiben. Ursprünglich hatte das W3C die Sprache zum Definieren von Metadaten konzipiert. Mittlerweile gilt RDF als ein grundlegender Baustein des „semantischen Webs“.

Die Technik ähnelt den Methoden zum Modellieren von UML-Klassendiagrammen und Entity-Relationship-Modellen. Häufig benutzte Aussagen über ein Objekt fasst sie zu sogenannten Ontologien zusammen, die sich über einen Namensraum (URI, Universal Resource Identifier) identifizieren lassen. Dies ermöglicht Programmen, Daten für den Menschen sinnvoll darzustellen. Eine Ontologie bezeichnet eine Sammlung von Begriffen, mit denen man Metadaten beschreibt, etwa Titel, Autor, Thema, Datum, Identifikator, Sprache, Ort einer Aufnahme und Kameratyp (bei Fotos). Gängige Ontologien sind etwa Dublin Core (Bibliothekswesen), IPTC (International Press Telecommunications Council) und Exif (Exchangeable Image File Format).