

Barrierefreie PDF-Dokumente mit Adobe Acrobat 7 Professional

Gute Tags, schlechte Tags

Marcus Hildebrandt, Maik Ludewig

Das Web ist voll davon: kaum noch ein Auftritt ohne PDF-Dokumente. Doch um deren barrierefreie Zugänglichkeit kümmert sich kaum jemand. Dabei lassen sich mit Adobe Acrobat Professional für jedermann zugängliche Dokumente erstellen. Mit Grundkenntnissen in Markup-Sprachen wie (X)HTML ist das Prinzip des „tagged PDF“ leicht zu verstehen und umzusetzen.

Spätestens seit dem Inkrafttreten der BITV (Barrierefreie Informationstechnik-Verordnung) im April 2002 nimmt die Zahl barrierefreier Informationsangebote im Internet rasch zu. Benachteiligte Personen wie Sehbehinderte oder Menschen mit motorischen Störungen haben dadurch die Möglichkeit, ihr Informationsrecht in der digitalen Welt wahrzunehmen. So lassen sich Inhalte mit Hilfe skalierbarer Schriften und Layouts, entsprechender Vorlesesoftware (Screen Reader) oder durch Verwendung einer Braille-Zeile erschließen. Was für HTML mittlerweile etabliert und praxiserprobt ist, wird für PDF-Dokumente jedoch oft vernachlässigt. Doch dafür gibt es spätestens seit Acrobat 5 keine Entschuldigung mehr.

Dokumente mit Mehrwert

Seit der Einführung von Adobe Acrobat 5 (PDF-Version 1.4, 2001) gibt es die Möglichkeit, ein PDF-Dokument mit einer Tag-Struktur zu versehen. Grundidee des Tagged PDF ist, Inhalt und Layout zu trennen und auf der logischen Struktur des Dokuments aufzusetzen. Das ermöglicht etwa ein leichteres Extrahieren der Textteile, das Hinterlegen von Informationen zum Lesefluss und so fort. [1]

Zusätzlich zu interpretierbaren Metadaten und Lesezeichen gewährleistet die Tag-Struktur, dass der Inhalt des PDF-Dokuments für assistive Technologien wie Screen Reader zugänglich ist. Ein Anwender kann ein „getaggetes“ PDF mittels einer Lesesoftware korrekt vorlesen lassen und sich dabei gezielt zu einzelnen Kapiteln oder Textabschnitten bewegen, Hyperlinks folgen, Erläuterungen zu Abkürzungen oder Akronymen erhalten und vieles mehr. Getaggte und barrierefrei aufbereitete Dokumente haben somit einen klaren Mehrwert für alle Benutzer.

Die Tag-Struktur eines PDFs erinnert in ihrer Syntax stark an (X)HTML beziehungsweise XML. Ähnlich einer (X)HTML-Seite wird der Inhalt linear und der Lesereihenfolge entsprechend getaggt. Elemente wie Überschriften, Absätze, Listen, Links und Tabellen

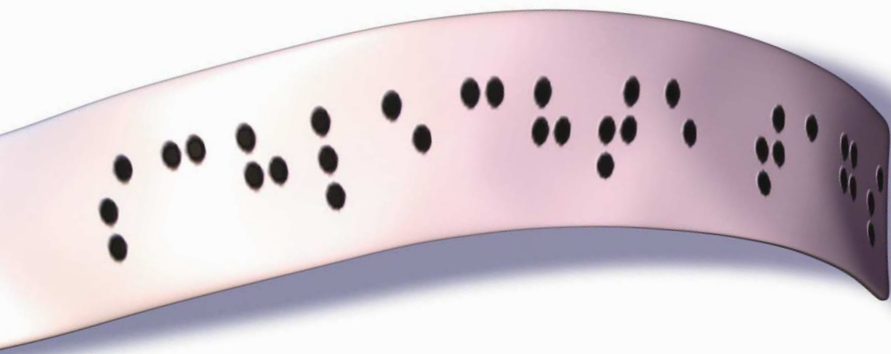
lassen sich mit dem von Adobe vordefinierten Tag-Satz auszeichnen, Sprachwechsel und Abkürzungen kennzeichnen. Besondere Bedeutung bei der Erstellung von barrierefreien PDF-Dokumenten kommt dem Alternativtext zu. Die Bereitstellung von sinnvollen Erweiterungstexten für wichtige Textabschnitte oder Dokumentenbereiche fördert das inhaltliche Verständnis und lässt Screen Reader nicht über Abkürzungen wie „usw.“ stolpern, indem sie diese als „uhsew“ vorlesen. Während Tags in einem (X)HTML-Dokument mit jedem beliebigen Texteditor bearbeitet werden können, ist die Professional Version von Adobe Acrobat für die Strukturierung und Nachbearbeitung eines PDFs unabdingbar. Alternative Werkzeuge existieren nicht. Mit Acrobat 7 Professional ist das Erstellen und Bearbeiten von Tags und Tag-Stämmen im Vergleich zu den Vorgängerversionen jedoch mittlerweile relativ komfortabel möglich.

An der Quelle ansetzen

Doch wie immer gilt: Vorbeugen ist besser als nachbessern. Der Aufwand für die Erstellung barrierefreier Dokumente ist umso geringer, je früher im Entstehungsprozess des PDFs man sich Gedanken um Barrierefreiheit macht.

XTRACT

- PDF wird zwar im Web als Dokumentenformat immer beliebter, doch kaum jemand achtet dabei auf barrierefreie Zugänglichkeit.
- Seit Acrobat 5 respektive PDF Version 1.4 ist eine Strukturierung durch Auszeichnungen (Tags) möglich, die auch assistive Technologien unterstützt.
- Ein barrierefreies PDF-Dokument ist nicht automatisiert zu erzeugen.



Ausgangspunkt ist die Wahl des passenden Text- oder Layoutprogramms. Wenn bereits im Quelldokument Strukturinformationen und Formatvorlagen verwendet werden, können Anwendungen wie Microsoft Word, Writer (Open Office) oder Adobe InDesign hieraus PDFs mit Tag-Stämmen generieren. Die Bearbeitungszeit für das manuelle Anlegen eines Strukturbaumes reduziert sich dadurch maßgeblich. Nach unseren Erfahrungen arbeitet übrigens OpenOffice Writer effizienter als Microsoft Word. Es setzt Tag-Bezeichnungen und die Strukturierung des Tag-Stamms besser um, und der Konvertierungsprozess nimmt weniger Zeit in Anspruch.

Derart erzeugte PDF-Dokumente enthalten eine Tag-Struktur, die zwar brauchbar ist, die aber dennoch ein Bearbeiter genau prüfen und in den meisten Fällen nachbearbeiten muss. Leere Tags, falsche Strukturierungen, zum Beispiel in Listen oder Tabellen, sind in nahezu allen automatisch getaggtten PDF-Dokumenten zu finden.

Sofern eine überschaubare Anzahl vorwiegend einfacher und gut strukturierter Quelldokumente in entsprechenden Formaten vorliegt, ist der Aufwand meist überschaubar. Anders sieht es aus, wenn keine Quelldateien mehr vorliegen und die vorhandenen PDF-Dokumente ursprünglich nicht für eine Veröffentlichung im Web konzipiert wurden, etwa Druckvorlagen für Broschüren, Flyer oder Faltblätter. Für solche Dokumente kann die nachträgliche barrierefreie Auszeichnung des Inhalts sehr aufwendig sein.

Altlasten aufarbeiten

Acrobat bietet die Möglichkeit, auch in bereits bestehenden PDF-Dateien Tags automatisch zu erzeugen und einen Tag-Stamm zu erstellen. Das klingt

zunächst gerade für große Mengen von Dokumenten attraktiv. Doch leider sind in der Praxis solche Tagstrukturen zum Großteil unbrauchbar, wenn man gehobene qualitative Maßstäbe anlegt.

Denn zahlreiche Layout- und Zierelemente (Schmuckgrafiken, Trennlinien, Rahmen, Textkästen et cetera) werden als Grafik ausgezeichnet und behindern den Lesefluss der Vorlesesoftware oder, falls doch inhaltlich relevant, beinhalten keine Alternativtexte. Oftmals sind Strukturelemente nicht korrekt interpretiert, die Inhalte falschen Tag-Typen zugewiesen, Überschriften und Listen als Absätze ausgezeichnet et cetera. Gleiches gilt für Tabellenelemente, die das maschinelle Tagging aufgrund fehlender Strukturierung im Ausgangsdokument nicht als solche erkennt und auszeichnet.

Gerade Tabellen sind aufgrund ihrer Informationsdichte sozusagen die „hohe Schule“ des Tagging, denn die Zuordnung in Spalten und Zeilen ist, sobald nicht mehr visuell präsentierbar, alles andere als trivial. Darum soll auch eine kleine Tabelle als Beispiel dienen,

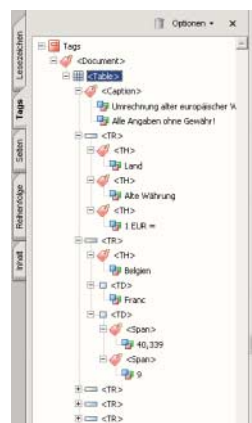
um ein paar Feinheiten der barrierefreien Auszeichnung zu erläutern.

Unter anderem folgende Tags sind von Adobe zur Auszeichnung von Tabellen vorgesehen:

- `<Table>` (Tag-Typ: Tabelle) entspricht dem gleichnamigen Tag in (X)HTML und beinhaltet als Container-Element die gesamte Tabellenstruktur.
- `<Caption>` (Tag-Typ: Beschriftung) kann im Sinne des (X)HTML-Attributs `summary` (Tabellenzusammenfassung) verwendet werden, um den Inhalt zu beschreiben. Im folgenden Beispiel sind die Tabellenüberschrift und die Fußzeile der Tabelle damit zusammengefasst.
- Die Strukturelemente (Tag-Typen: Tabellenzeile, Tabellenüberschriftenzeile und Tabellendatenzeile) `<TR>`, `<TH>` und `<TD>` entsprechen der (X)HTML-Syntax.

Wie der Screen Reader liest

Die Lesereihenfolge von getaggtten Informationen wird über die Reihenfolge im Tag-Stamm bestimmt. In (X)HTML wird die Tabellenzusammenfassung im öffnenden Table-Tag definiert und infolgedessen als erste Information durch den Screen Reader vorgelesen. Damit der Nutzer sofort einen Überblick über die Inhalte der Tabelle erhält, ist es also auch in einem PDF-Dokument sinnvoll, nach dem `<Table>`- direkt einen

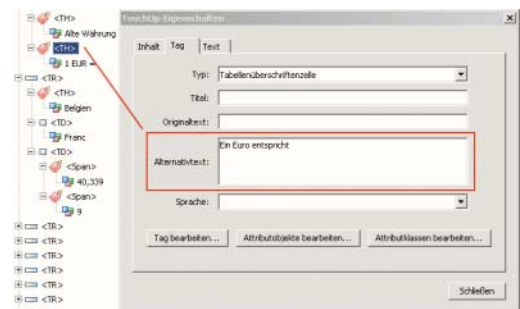


Unrechnung alter europäischer Währungen in Euro

Land	Alte Währung	1 EUR =
Belgien	Franc	40,3399
Deutschland	D-Mark	1,95538
Finnland	Mark	5,94573
Frankreich	Franc	6,55957
Holland	Gulden	2,20371
Irland	Pfund	0,787564
Italien	Lira	1936,27
Luxemburg	Franc	40,3399
Österreich	Schilling	13,7603
Portugal	Escudo	200,482
Spanien	Peseta	166,386

Alle Angaben ohne Gewähr! <Caption>

Die richtigen Tags und eine lineare Strukturierung der Daten sorgen für Zugänglichkeit (Abb. 1).



Verständlicher Alternativtext sorgt für sinnvolle Sprachausgabe (Abb. 2).

<Caption>-Tag in die Struktur zu integrieren. Als Inhalt für diesen Tag dient im Beispiel zum einem die Überschrift, zum anderen die Fußzeile.

Um beurteilen zu können, ob der Zelleninhalt verständlich ist, muss man wissen, wie ein Screen Reader Inhalte liest. Meist sind erfahrene Screen-Reader-Nutzer darin geübt, selbst aus unverständlich vorgelesenen Inhalten die richtigen Informationen herauszuhören. Mit wenig Mehraufwand kann der Bearbeiter hier über die sinnvolle Zuweisung von Alternativtexten für einen angenehmeren Lesefluss und eine verständlichere akustische Wiedergabe sorgen.

Die erste Zeile der Beispieltabelle enthält drei Überschriftenzellen, wobei die letzte Zelle auf der rechten Seite den Gegenwert der alten Währung für einen Euro wiedergibt.

An dieser Stelle entstehen drei Leseprobleme:

- Die Zahl „1“ wird nicht korrekt vorgelesen („eins“ anstatt – wie sinnvoll erforderlich – „ein“).
- Die Abkürzung „EUR“ für Euro wird unverständlich gelesen („euer“ anstatt „Euro“).
- Das Gleichheitszeichen („=“) wird durch den Screen Reader im Wortlaut „gleich“ vorgelesen.

So liest der Screen Reader vor: „Eins Euer gleich“. Die Lösung besteht im Ersetzen der in der Überschriftenzelle getagten Auswahl durch einen sinnvollen Alternativtext.

Grundsätzlich wird der tatsächliche Inhalt eines Tag-Containers durch die Einbindung eines Alternativtextes übersteuert. Durch das Einfügen des Alternativtextes „Ein Euro entspricht“ ist die Bedeutung der Überschriftenzelle somit wesentlich verständlicher.

Die Auszeichnung von Sprachwechseln dient der korrekten Aussprache von Inhalten, die von der vorherrschenden Dokumentensprache abweichen. Ab Version 7 liest der Screen Reader Jaws fremdsprachlich ausgezeichnete Bereiche nun auch in der jeweiligen Sprache vor.

An mehreren Stellen der Tabelle innerhalb der Spalte „Alte Währung“ muss eine Sprachauszeichnung erfolgen. Dem Zelleninhalt wird dafür das entsprechende Sprachattribut in den Eigenschaften des zugehörigen Container-Tag (hier <TD>) zugewiesen. In der Syntax vergleichbar ist die Verwendung von `xml:lang="ISO-Kürzel"` in (X)HTML.

Sprachwirrwar beheben

Was für einen sehenden User gut verständlich ist, kann für den Nutzer einer Screen-Reader-Software möglicherweise unverständlich sein.

In der letzten Spalte der Tabelle sind die Gegenwerte der alten Währung für einen Euro aufgeführt. Zur besseren Verständlichkeit während des Vorlesens ist es daher vorteilhaft, wenn zu dem Zahlenwert noch einmal die jeweilige Währung genannt wird. Das kann – wie zuvor bereits erklärt – über die Zuweisung eines entsprechend erweiterten Alternativtextes erfolgen, den der Screen Reader dann anstelle des tatsächlichen Zelleninhalts vorliest.

Ein Problem ergibt sich, wenn die Währungsbezeichnung fremdsprachlich ausgezeichnet werden muss: In einer Zelle befinden sich jetzt ein Währungsbetrag, den der Reader in der vorherrschenden Dokumentensprache (in diesem Fall deutsch) vorlesen soll,

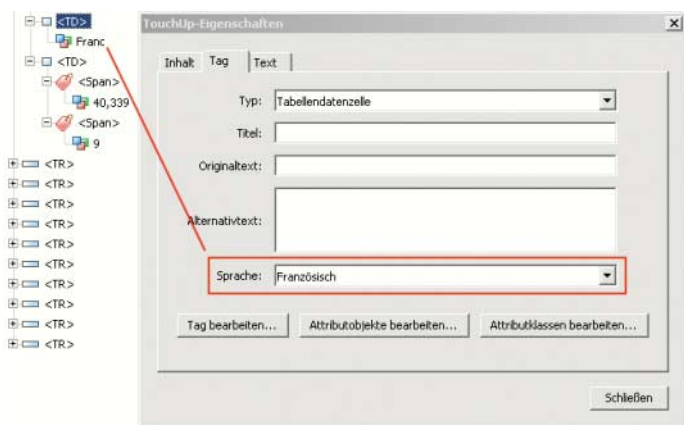
gleichzeitig ist der Alternativtext dieser Zelle um eine Währungsbezeichnung mit französischem Sprachattribut erweitert.

Hier lässt sich also nicht mit der Festlegung eines Sprachattributs für die gesamte Datenzelle arbeiten, da dann sowohl Währungsbetrag als auch Währungsbezeichnung in der gleichen Sprache vorgelesen würden. Zur Lösung des Problems legt man innerhalb der Datenzelle zwei weitere Container-Tags an und zeichnet sie unabhängig voneinander mit Sprachattributen aus. Analog zu (X)HTML bietet sich hier ein -Tag (Tag-Typ: Bereich) an.

Daraus ergibt sich jedoch eine weitere Problematik: Der eigentliche Inhalt dieser Datenzelle besteht nur aus dem Währungsbetrag, was bedeutet, dass für den zweiten -Tag, der die Währungsbezeichnung aufnehmen soll (hier „Franc“) kein regulär „tagbarer“ Inhalt zur Verfügung steht.

Da ein Alternativtext den getagten Inhalt eines Tag-Containers übersteuert, kann der vorhandene Inhalt (also der Währungsbetrag) aufgeteilt werden, sodass sich beide -Tags mit Inhalt füllen lassen. Visuell ändert sich dadurch für den sehenden Nutzer im Dokument nichts.

Was zuerst umständlich erscheinen mag, ist in der Praxis zwingend erforderlich, aber zugleich recht simpel umzusetzen. Der Grund dafür ist, dass sich leere Tags nicht mit Alternativtext versehen lassen, da dies einen Fehler im Ausgabebelegbericht von Acrobat erzeugen würde. Auch der Screen Reader würde an dieser Stelle nichts vorlesen. Es muss sich also mindestens ein tagbares Zeichen innerhalb des jeweiligen Container-Tags befinden.



Dem Zelleninhalt wird das Sprachattribut „Französisch“ zugewiesen (Abb. 3).

Land	Alte Währung	1 EUR =
	Franc	40,3399
	D-Mark	1,95538
	Mark	5,94573
	Franc	6,55957
	Gulden	2,20371
	Pfund	0,787564
	Lira	1936,27
Luxemburg	Franc	40,3399
Österreich	Schilling	13,7603
Portugal	Escudo	200,482
Spanien	Peseta	166,386

Der -Tag wird durch eine Aufteilung des Zelleninhalts befüllt (Abb. 4).

Land	Alte Währung	1 EUR =
Belgien	Franc	40,3399
Frankreich	Franc	6,55957
Luxemburg	Franc	40,3399
Österreich	Schilling	13,7603

Einem Tag werden ein Alternativtext und ein Sprachattribut zugewiesen (Abb. 5).

Im ersten ``-Tag wird daher der Währungsbetrag bis auf die letzte Ziffer getaggt, in das Feld für den Alternativtext hingegen der komplette Betrag als Zahl eingefügt. Der zweite ``-Tag bekommt die letzte Zahl des Währungsbetrags (die „9“) als Inhalt. Nun kann man diesem ``-Tag auch den für den Screen Reader maßgeblichen Alternativtext zuweisen, in diesem Fall die Währungsbezeichnung „Franc“. Hinzu kommt die französische Sprachauszeichnung, um ein korrektes Vorlesen durch den Screen Reader zu ermöglichen.

Überprüfung auf Barrierefreiheit

Der integrierte Ausgabehilfebericht in Acrobat (Menü „Erweitert->Ausgabe-hilfe->Vollständige Prüfung“) ermöglicht das Aufspüren potentieller Barrieren innerhalb der Dokumentstruktur und ist ein zentrales Werkzeug zur Erstellung barrierefreier Dokumente. Der Prüfmechanismus liefert Aussagen zu fehlenden Alternativtexten für Bilder und Grafiken, unzugänglichen Verlinkungen, Elementen, die nicht in die Tag-Struktur integriert sind, und vieles mehr. Im optimalen Fall sollte die Prüfung keine Fehler mehr ergeben. Als letzter Schritt bei der Qualitätssicherung ist ein Lesetest des gesamten Dokuments mit einer verbreiteten Screen-Reader-Software empfehlenswert.

Ein barrierefreies PDF-Dokument lässt sich derzeit noch nicht gegen einen öffentlich definierten Standard validieren. Vielmehr sind hier der Sachverstand und die Sensibilität für die Zugänglichkeit seitens des Bearbeiters gefragt, um den Inhalt zu strukturieren und für eine möglichst große Benutzergruppe barrierefrei aufzubereiten.

Fazit

Das Beispiel zeigt, dass man auch eine Datentabelle so aufbereiten kann, dass der Inhalt über die akustische Ausgabe eines Screen Readers gut zu erschließen ist. Es sollte jedoch auch deutlich geworden sein, dass weitaus mehr als das automatisierte Erstellen von Strukturinformationen notwendig ist, um vollständige Barrierefreiheit innerhalb eines PDF-Dokuments zu gewährleisten. Das Hinzufügen sinnvoller Alternativtexte, die Herstellung einer korrekten Lesereihenfolge, letztendlich die Entscheidung, welche Informationen wichtig sind – das sind Aufgaben, für die es einen geschulten Redakteur braucht.

Neben der notwendigen Fachkenntnis im Umgang mit Adobe Acrobat und Markup-Sprachen bedarf es vor allem der Sensibilisierung für die Thematik barrierefreier Webinhalte bei Anbietern und Autoren von PDF-Dokumenten. (JS)

MARCUS HILDEBRANDT

arbeitet als Consultant und Dozent in Sachen Barrierefreiheit bei der Babiel GmbH in Düsseldorf.

MAIK LUDEWIG M. A.

ist Projektmanager mit Schwerpunkt Informationsmanagement bei der Babiel GmbH.

Literatur

- [1] Adobe Systems Incorporated; PDF Reference Fifth Edition; Version 1.6; online: http://partners.adobe.com/public/developer/pdf/index_reference.html
- [2] Adobe, www.adobe.com/accessibility/index.html

