

Erkennungstiefe bei Identifikations- und Validierungstools

Bei der Benutzung von Software zur Identifikation oder Validierung von Dateiformaten stößt man in der Praxis immer wieder auf Fälle, bei denen unterschiedliche Werkzeuge abweichende Ergebnisse zum Format oder zur Validität einer Datei ergeben.

Die Grundlage der Identifikation und Validierung bilden meist Dateiformatspezifikationen. Diese können recht umfangreich und komplex sein, sind nicht immer eindeutig und lassen somit Raum für Interpretationen. Dies kann dazu führen, dass eine bestimmte Eigenschaft eines Dateiformats hinsichtlich der Bedeutung für die Validität unterschiedlich interpretiert wird. Andererseits können sich die Tools auch im Umfang und der Genauigkeit der durchgeführten Prüfungen unterscheiden. Insbesondere bei Anwendungen, die verschiedene Werkzeuge unter einer einheitlichen Oberfläche integrieren, kann dies zu Irritationen führen, wenn sich die Validierungstiefe der verwendeten Werkzeuge unterscheidet. Dies mag umso schwerer wiegen, da sich diese Anwendungen häufig an unerfahrene Anwender richten, die mit Hilfe der Anwendung komfortabel zu zuverlässigen Ergebnissen kommen wollen.

Beispiel Validierung: PDF/A-Validierung

Anhand der Ergebnisse von zwei Programmen zur Validierung von PDF/A-Dateien, soll die unterschiedliche Interpretation aufgezeigt werden (Abbildung 1).¹ Die Validierung mit dem Preflight-Werkzeug von Adobe Acrobat Pro DC (v2015) findet keine Probleme (oben).² Der Open Source PDF/A Validator veraPDF (v1.2.1) kommt zu einem gegenteiligen Ergebnis (unten).³ Die Auflösung auftretender Widersprüche setzt oftmals ein tiefgreifendes Fachwissen voraus und lässt sich selten Verallgemeinern.

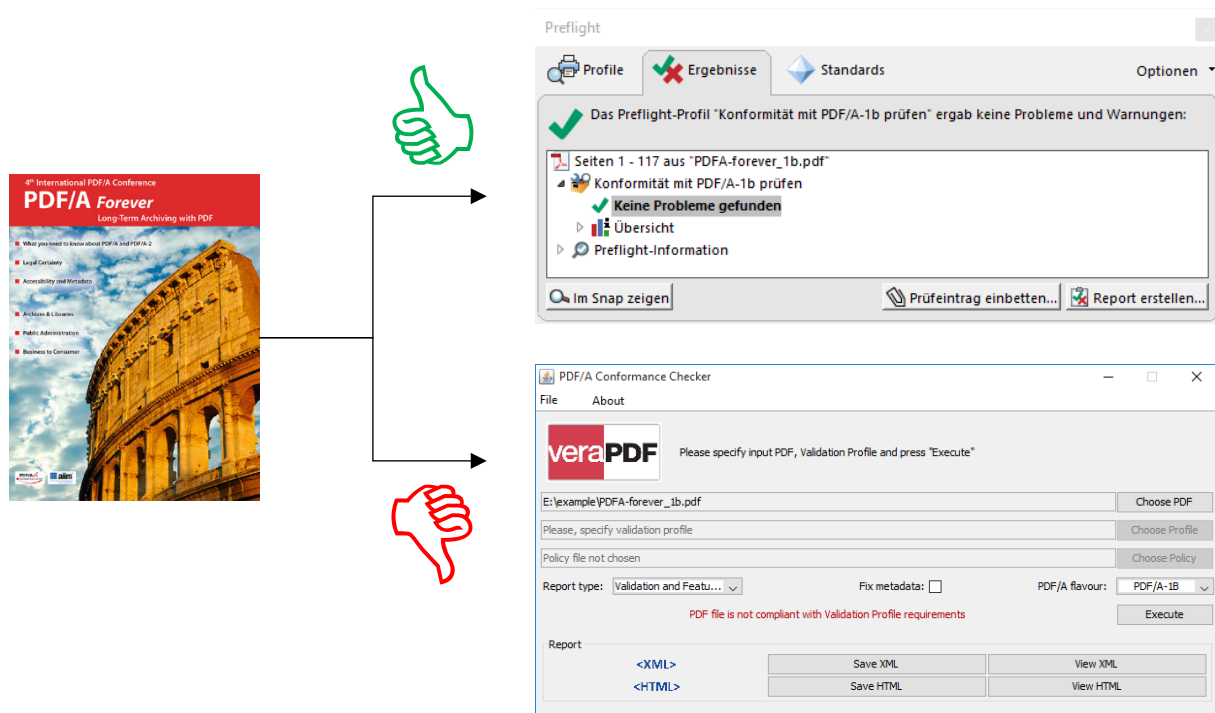


Abbildung 1: Die Validierungsergebnisse eines PDF/A-Dokuments mit Adobe Acrobat Pro DC (oben) und veraPDF (unten).

¹ Als Beispiel dient das Dokument im Format PDF/A-1b *PDF/A Forever* herausgegeben vom PDF/A Competence Center.

PDF/A Competence Center: PDF/A Forever. https://www.pdfa.org/wp-content/untill2016_uploads/2011/08/PDFA-forever_1b.pdf

² Adobe Systems: Adobe Acrobat. <https://acrobat.adobe.com/de/de/acrobat.html>

³ veraPDF consortium: veraPDF. <http://verapdf.org/>

Beispiel Validierung: KOST-Val

KOST-Val⁴ ist eine von der Koordinationsstelle für die dauerhafte Archivierung elektronischer Unterlagen (KOST) entwickelte Anwendung zur Validierung von Dateiformaten die unterschiedliche, bereits existierende Validatoren unter einer Oberfläche vereint. Der beschriebenen Problematik widersprüchlicher Analyseergebnisse begegnet die Anwendung durch die Möglichkeit der mehrstufigen Validierung von PDF/A-Dateien durch unterschiedliche Validatoren. Für andere Dateiformate steht diese Möglichkeit allerdings nicht zur Verfügung. So werden die Dateiformate TIFF mittels JHOVE und JPEG mittels Bad Peggy analysiert.⁵ Diese Programme besitzen jedoch eine unterschiedliche Prüfungstiefe. So werden beispielsweise in die Dateien eingebundene Farbprofile nur von Bad Peggy in die Prüfung einbezogen, wie folgendes Beispiel zeigt (Abbildung 2):⁶

Eine TIFF- und JPEG-Datei werden mittels KOST-Val analysiert und als valide eingestuft.

Valid:

Validierung: TIFF -> E:\example\kost-val\no-icc\example.tif
Validierung: JPEG -> E:\example\kost-val\no-icc\example.jpg

An diese Beispieldateien wird ein Farbprofil mit einer ungültigen Profiklasse angefügt und erneut geprüft. Aufgrund der unterschiedlichen Validierungstiefe der verwendeten Validatoren, wird nur die JPEG-Datei als ungültig ausgewiesen.

Invalid:

Validierung: JPEG -> E:\example\kost-val\icc\example.jpg
A) Erkennung und BadPeggy Die BadPeggy-Validierung wurde nicht bestanden.
D) Andere Probleme Der Fehler wurde noch nicht uebersetzt. Bitte der KOST melden. Original-Message: Unknown profile class.

Valid:

Validierung: TIFF -> E:\example\kost-val\icc\example.tif

Abbildung 2: Ergebnisse von Validierungen mit KOST-Val

Beispiel Identifikation: DROID

DROID⁷ ist eine von The National Archives (TNA) entwickelte Anwendung zur Identifikation von Dateiformaten. Zur Identifikation werden die in der PRONOM technical registry gesammelten Signaturen verwendet.⁸ Oftmals finden sich die zur Identifikation herangezogenen Signaturen am Anfang oder Ende einer Datei. So ist in den Voreinstellungen des Programms die maximale Anzahl von gescannten Bytes am Anfang und Ende auf 65536 Bytes festgelegt.⁹ Nicht zuletzt aus Performancegründen mag diese Begrenzung sinnvoll erscheinen, bringt in der Praxis jedoch im Einzelfall

Maximum bytes to scan at the start and end of files.
A negative value means unlimited scanning.

Abbildung 3: Voreinstellung des zu analysierenden Bereichs in DROID

Resource	Extension	Ids	PUID	Method
E:\example\example.accdb	accdb			
E:\example\example.accdb	accdb		fnt/275	Signature

Abbildung 4: Analyseergebnisse von DROID einer Microsoft Access Datenbank mit eingeschränktem Scanbereich (65536 Bytes, oben) und nicht limitiertem Scanbereich (unten)

⁴ Koordinationsstelle für die dauerhafte Archivierung elektronischer Unterlagen: KOST-Val. https://kost-ceco.ch/cms/kost_val_de.html

⁵ Detaillierte Informationen zur dualen PDF/A-Validierung und den verwendeten Validatoren finden sich im Anwendungshandbuch zu KOST-Val unter <http://github.com/KOST-CECO/KOST-Val/releases/latest>

⁶ Die Validierung wurde durchgeführt mit KOST-Val 1.7.9

⁷ The National Archives: Download DROID: file format identification tool. <http://www.nationalarchives.gov.uk/information-management/manage-information/preserving-digital-records/droid/>

⁸ ebenda

⁹ Diese Voreinstellung wird auch an anderer Stelle empfohlen. Vgl. Stephanie Kortyla, Christian Treu: Nutzen und Grenzen der Formatidentifizierung beim Preservation Planning. Seite 33. https://www.sg.ch/content/dam/sgch/kultur/staatsarchiv/auds-20181/dokumentationen-camps/10-3_Kortyla_Treu.pdf in Arbeitskreis Archivierung von Unterlagen aus digitalen Systemen (AUDS), 21. Tagung. <https://www.sg.ch/kultur/staatsarchiv/Spezialthemen-/auds/2017.html>

unzureichende Analyseergebnisse mit sich, sodass für die verlässlichere Analyse nur zu einem vollständigen Scan der zu untersuchenden Dateien geraten werden kann.¹⁰

Fazit

Bei der Benutzung von Validierungswerkzeugen ist es notwendig, die Limitationen und Schwachstellen der jeweiligen Programme zu kennen, um im Rahmen der eigenen Anforderungen entsprechend darauf reagieren zu können. Insbesondere bei Anwendungen, die unterschiedliche Werkzeuge kombinieren, muss darauf geachtet werden, welche Prüfungen von welchem Programm in das Endergebnis einfließen. Es ist unabdingbar die aufgetretenen Fehler einordnen zu können. Fehler und Unstimmigkeiten sollten dem Entwickler des Validators und der Fachwelt mitgeteilt werden. Abweichende Validationsergebnisse sollten dokumentiert werden.

¹⁰ Ein vollständiger Scan lässt sich durch die Vorgabe eines negativen Wertes (z. B. -1) in den Voreinstellungen von DROID erzwingen. Die Identifikation im Beispiel wurde durchgeführt mit DROID 6.4. Als Testdatei dient eine leere Desktopdatenbank, erstellt mit Access 2013.