



Anwendungshandbuch

Inhalt

1	Vorwort	2
2	Systemvoraussetzungen	3
3	Offene Punkte / Feedback	3
4	Einleitung duale PDF/A-Validierung	3
5	Diagnosedaten.....	4
5.1	PDF-Diagnosedaten.kost-val.xml	4
5.2	KaD-Diagnosedaten.kost-val.xml.....	4
6	Installation.....	5
6.1	Installation von KOST-Val.....	5
6.2	Optionale Installation von Drittapplikationen.....	5
7	Konfiguration des KOST-Val.....	7
7.1	Bestandteile der Konfigurationsdatei "kostval.conf.xml"	7
7.2	"KaD_SignatureFile_V71.xml" und "jhove.conf"	9
8	Ressourcen des KOST-Val.....	10
9	Validierung starten.....	11
9.1	Validierung starten mittels "KOST-Val_de.exe"-GUI.....	11
9.2	Validierung manuell starten	13
10	Validierung auswerten	15
11	Beschreibung der Validierungsschritte	16
11.1	TIFF	16
11.2	SIARD	17
11.3	PDF/A	18
11.4	JP2.....	19
11.5	JPEG.....	19
11.6	SIP	20
12	Urheberrecht.....	21
12.1	PDF/A Manager von PDFTron.....	22
12.2	3-Heights™ PDF/A Validator API-Lizenz.....	22
13	Anhang	24
13.1	Beschreibung der Konfigurationsdatei "kostval.conf.xml"	24
13.2	Programmaufbau	28

1 Vorwort

KOST-Val ist eine java-basierte Anwendung zur Validierung von Aufbau und Inhalt von TIFF-Dateien (*Tagged Image File Format*), SIARD-Dateien (*Software Independent Archiving of Relational Databases*), PDF/A-Dateien, JP2-Dateien (JPEG 2000; *Joint Photographic Experts Group 2000*) JPEG-Dateien sowie von sogenannter Submission Information Package (SIP) zur Ablieferung von digitalen Informationen. Diese Anwendung steht unter der GPL3+ Lizenz und wird durch die KOST der Öffentlichkeit quelloffen zur Verfügung gestellt. KOST-Val stützt sich auf unveränderte Komponenten anderer Hersteller, welche direkt im Quellcode von KOST-Val eingebunden sind. Die Benutzer von KOST-Val sind gehalten, die Lizenzbestimmungen all dieser Komponenten zu befolgen. Ausführliche Informationen sind im Kapitel 12 ersichtlich.

KOST-Val erfüllt die im Folgenden beschriebenen Anforderungen.

TIFF-Validierung: KOST-Val liest ein TIFF und validiert mit Hilfe von Jhove den Aufbau und Inhalt sowie die zentralen Eigenschaften wie z.B. Komprimierung, Farbraum und Multipage. Die Eigenschaften sind konfigurierbar.

SIARD-Validierung: KOST-Val liest ein SIARD (eCH-0165 v1¹) und validiert den Aufbau und Inhalt wie z.B. Struktur, Header- und Content-Validierung.

PDF/A-Validierung: KOST-Val liest ein PDF respektive PDF/A (ISO 19005-1 und 19005-2) und validiert mit Hilfe des 3-Heights™ PDF/A Validator von PDF-Tools oder auch des PDF/A-Manager von PDFTron den Aufbau und Inhalt der PDF-Datei. KOST-Val gliedert die verschiedenen Fehlermeldungen in Hauptgruppen wie z.B. Schriften, Grafiken und Metadaten. Im Lieferumfang von KOST-Val ist einzig eine eingeschränkte² Version von 3-Heights™ PDF/A Validator von PDF-Tools enthalten.

JP2-Validierung: KOST-Val liest ein JP2 (ISO 15444) und validiert mit Hilfe von Jpylyzer den Aufbau und Inhalt.

JPEG-Validierung: KOST-Val liest ein JPEG (ISO 10918-1) und validiert mit Hilfe von BadPeggy den Aufbau und Inhalt.

SIP-Validierung: KOST-Val liest ein SIP (eCH-0160 v1³ sowie BAR v1⁴ und v4⁵) und validiert die Muss Punkte aus der SIP Spezifikation. Die einzelnen validierten Punkte werden in Gruppen wie z.B. Verzeichnisstruktur, Schema- und Prüfsummenvalidierung eingegliedert. Zuvor wird eine Formatvalidierung durchgeführt.

Die Resultate (inklusive Meldungen zu Inkonsistenzen oder Fehler) werden pro Schritt ausgegeben und in eine Validierungs-Logdatei geschrieben.

Die einzelnen Validierungsschritte / Prüfungen werden nacheinander ausgeführt. Wo möglich, wird die Validierung auch bei Fehlern weiter fortgesetzt, um die Anzahl von Korrekturzyklen zu reduzieren.

¹ Die Spezifikation kann von der eCH Website heruntergeladen werden:

<http://www.ech.ch/vechweb/page?p=dossier&documentNumber=eCH-0165> .

² Die Einschränkung bezieht sich hauptsächlich auf den Durchsatz von bis zu 72'000 Seiten pro Jahr. Mehr Informationen sind in der Lizenz in Kapitel 12.2 ersichtlich. Diese Einschränkung kann aufgehoben werden, in dem eine 3-Heights™ PDF/A Validator API Lizenz bei PDF-Tools erworben und mit dem Lizenzmanager aktiviert wird.

³ Die Spezifikation kann von der eCH Website heruntergeladen werden:

<http://www.ech.ch/vechweb/page?p=dossier&documentNumber=eCH-0160> .

⁴ Die Spezifikation kann von der Seite "Ablieferungsschnittstelle" der BAR-Website heruntergeladen werden: <http://www.bar.admin.ch/themen/00876/00877/>.

⁵ Die Spezifikation kann von der Seite "Ablieferungsschnittstelle" der BAR-Website heruntergeladen werden: <http://www.bar.admin.ch/themen/00876/00877/01561/>.

Detailliertere Informationen zu den einzelnen Formaten und Validierungsschritte sind im Anhang ersichtlich.

2 Systemvoraussetzungen

Microsoft Windows 98 und neuer

Mindestens 128 MB RAM (512 MB oder mehr empfohlen)

Mindestens 20 GB Festplattenspeicher

Java Runtime Environment (JRE) Version 6 respektive 1.6

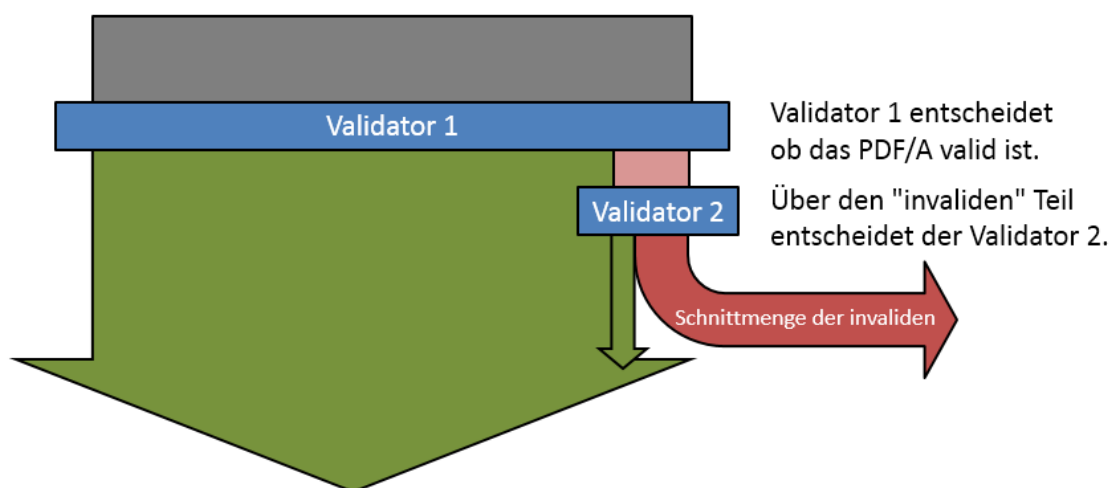
3 Offene Punkte / Feedback

Die offenen Punkte, von Bugs respektive Fehler bis hin zu Ergänzungswünschen und Fragen sind auf der Entwicklungsplattform GitHub unter Issues ersichtlich (<https://github.com/KOST-CECO/KOST-Val/issues>) und können an kost-val@kost-ceco.ch gemeldet werden.

Diese Liste kann und soll durch jedermann erweitert werden und wird durch das Entwicklerteam bearbeitet.

4 Einleitung duale PDF/A-Validierung

Für PDF/A bietet KOST-Val die Möglichkeit einer dualen Validierung. Dabei wird eine PDF/A-Datei zunächst durch einen ersten Validator geprüft. Bei invalidem Resultat folgt eine Prüfung durch einen zweiten Validator. Die PDF/A-Datei gilt als valid, wenn mindestens einer der Validatoren sie als valid identifiziert, und als invalid, wenn beide Validatoren sie als invalid identifizieren.⁶



Die duale PDF/A-Validierung darf nur angewendet werden, wenn das Archiv es zulässt, dass potenziell invalide PDF/A-Dateien übernehmen werden dürfen. Sollte dies nicht der Fall sein, dann sollte auf die duale PDF/A-Validierung verzichtet werden.

⁶ Die duale Validierung kann nur mit qualitativ hochstehenden PDF/A-Validatoren in diesem Sinne durchgeführt werden. Diese hohen Anforderungen erfüllen unter anderem die neusten Versionen von 3-Heights™ PDF/A Validator von PDF-Tools und PDF/A-Manager von PDFTron.

Für die duale Validierung müssen sowohl 3-Heights™ PDF/A Validator von PDF-Tools als auch PDF/A-Manager von PDFTron lizenziert und installiert sein. Im Lieferumfang von KOST-Val ist einzig eine eingeschränkte Version von 3-Heights™ PDF/A Validator von PDF-Tools enthalten. Wenn nur ein Validator vorhanden ist, wird automatisch eine einfache Validierung durchgeführt.

Konzeptionelle Grundlage für die duale Validierung ist die Feststellung, dass selbst qualitativ hochstehende PDF/A-Validatoren zu unterschiedlichen Resultaten kommen können. Dies liegt einerseits daran, dass der eigentliche PDF/A-Standard ein Set von anderen Standards einschliesst, welche in den Validatoren nicht zwingend bis in alle Details implementiert sind. Andererseits sind gewisse Vorgaben des Standards so formuliert, dass sie legitimerweise auf verschiedene Arten implementiert werden können. Dass sämtliche relevanten Tools die Spezifikation einheitlich und vollständig implementieren, bleibt vorerst Zukunftsmusik. Deshalb bietet KOST-Val als Zwischenlösung die duale Validierung an.

5 Diagnosedaten

KOST-Val erstellt in einigen Fällen verschiedene Diagnosedaten. Diese sind nachfolgende beschrieben und der Pfad zu den Diagnosedaten kann in der Konfiguration (siehe auch Kapitel 7.1.1) bei Bedarf angepasst werden.

5.1 PDF-Diagnosedaten.kost-val.xml

Während der Verwendung der dualen PDF/A-Validierung sammelt KOST-Val bei Uneinigkeit der Validatoren in der Datei "PDF-Diagnosedaten.kost-val.xml" entsprechende PDF-Diagnosedaten, die für die Verbesserung der Tools und des Standards hilfreich sein können. Es wird nebst dem vollständigen Dateipfad, die PDF/A-Version mit der gewünschten Konformität und die uneinigen Fehlermeldungen gesammelt.

Wir sind Ihnen dankbar, wenn Sie diese PDF-Diagnosedatei bei Gelegenheit der KOST-Geschäftsstelle (kost-val@kost-ceco.ch) zusenden könnten.

5.2 KaD-Diagnosedaten.kost-val.xml

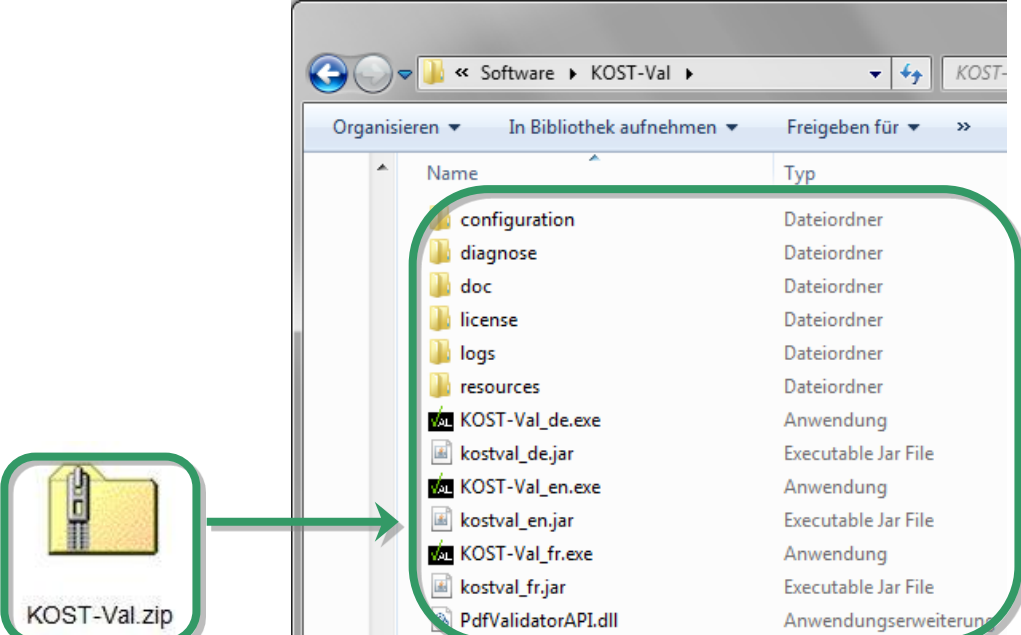
Während der Verwendung der SIP-Validierung sammelt KOST-Val Informationen über die vorhandenen KaD-Dateiformate und deren Anzahl in "KaD-Diagnosedaten.kost-val.xml". Dies bietet die Möglichkeit, zu erfahren welche Dateiformate in welcher Häufigkeit im content-Ordner der validierten SIPs enthalten sind und kann entsprechend der Weiterentwicklung von KOST-Val (zusätzliche Dateiformate) als auch eines systematischen PreservationPlannings mit einem Risikobasierten Ansatz verwendet werden.

Wir sind Ihnen dankbar, wenn Sie diese KaD-Diagnosedatei auf Anfrage der KOST-Geschäftsstelle (kost-val@kost-ceco.ch) zusenden könnten.

6 Installation

6.1 Installation von KOST-Val

1 KOST-Val herunterladen und in das gewünschte Verzeichnis entpacken.



Name	Typ
configuration	Dateiordner
diagnose	Dateiordner
doc	Dateiordner
license	Dateiordner
logs	Dateiordner
resources	Dateiordner
KOST-Val_de.exe	Anwendung
kostval_de.jar	Executable Jar File
KOST-Val_en.exe	Anwendung
kostval_en.jar	Executable Jar File
KOST-Val_fr.exe	Anwendung
kostval_fr.jar	Executable Jar File
PdfValidatorAPI.dll	Anwendungserweiterung

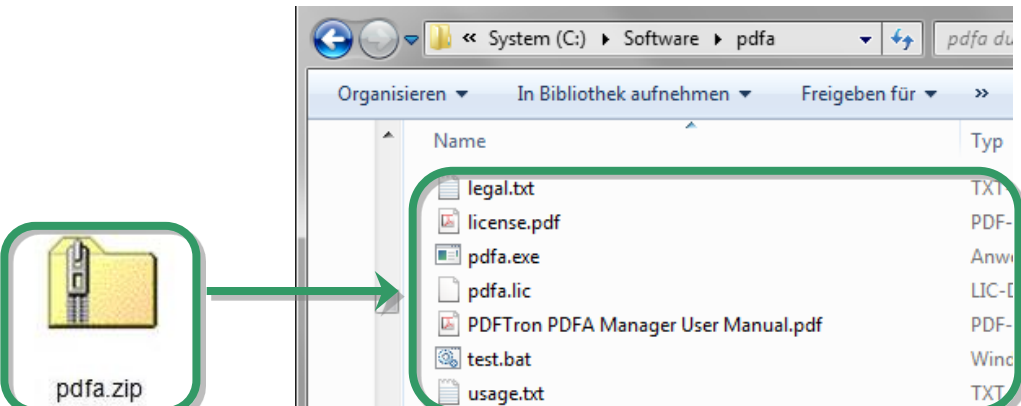
6.2 Optionale Installation von Drittanwendungen

6.2.1 Installation von PDF/A Manager von PDFTron

! Für die Durchführung der PDF/A-Validierung wurde auch die indirekte Verwendung (via Konsolenaufruf) der kommerziellen Software PDF/A Manager von PDFTron implementiert. Zu Testzwecken kann eine Testversion heruntergeladen werden. Sollte der PDF/A-Manager von PDFTron nicht installiert sein, wird der integrierte 3-Heights™ PDF/A Validator von PDF-Tools verwendet.

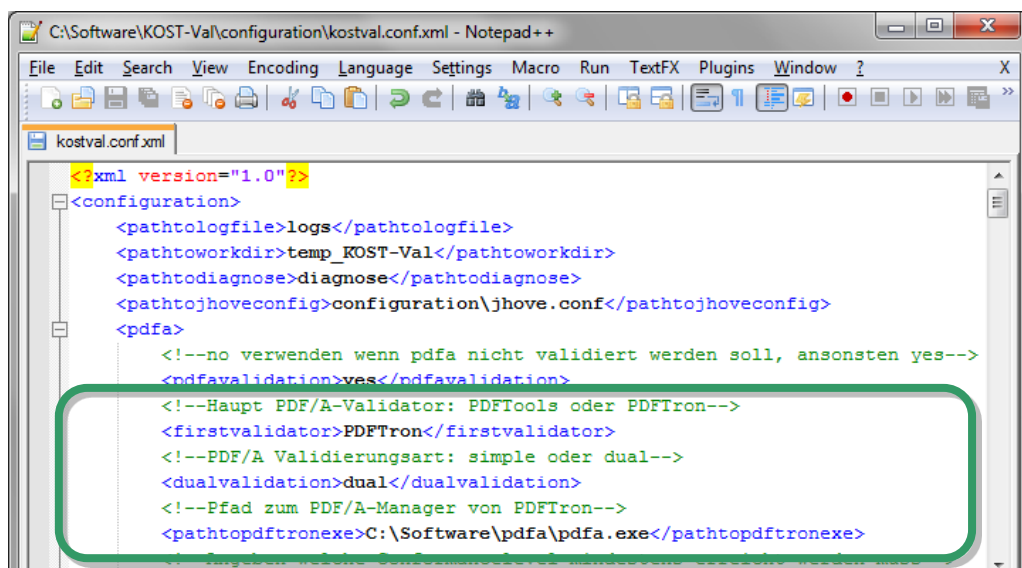
1 Command-line-Variante des PDF/A Managers (ZIP-File) unter <http://www.pdftron.com/downloads.html> herunterladen.

2 Den PDF/A Manager in ein beliebiges Verzeichnis entpacken.



Name	Typ
legal.txt	TXT
license.pdf	PDF-
pdfa.exe	Anw
pdfa.lic	LIC-
PDFTron PDF/A Manager User Manual.pdf	PDF-
test.bat	Wind
usage.txt	TXT

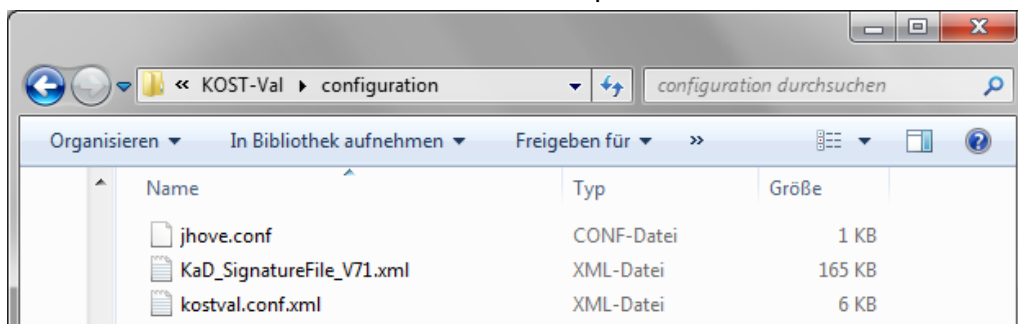
- 3 Die Konfigurationsdatei "KOST-Val\configuration\kostval.conf.xml" muss mit dem entsprechenden Pfad zu pdfa.exe ergänzt und anschliessend kann die duale PDF/A-Validierung bei Bedarf eingeschaltet werden. Zudem sollte der Hauptvalidator auf PDFTron umgestellt werden, ausser wenn die Einschränkung im 3-Heights™ PDF/A Validator von PDF-Tools aufgehoben wurde. Für diese Modifikationen sollte ein XML-Editor oder Notepad ++⁷ verwendet werden.



⁷ Die Portable Notepad ++ Applikation kann unter http://portableapps.com/de/apps/development/notepadpp_portable heruntergeladen werden.

7 Konfiguration des KOST-Val

Sämtliche Konfigurationen des KOST-Val können im Unterordner "configuration" in der Konfigurationsdatei "kostval.conf.xml" vorgenommen werden. Für Modifikationen an XML-Dateien sollte ein XML-Editor oder Notepad ++⁸ verwendet werden.



Im Ordner "configuration" sind auch die Dateien "jhove.conf" und "KaD_SignatureFile_V71.xml" abgelegt. Diese sind im Kapitel 7.2 beschrieben.

7.1 Bestandteile der Konfigurationsdatei "kostval.conf.xml"

Die Konfigurationsdatei "kostval.conf.xml" ist in verschiedenen Teilen aufgebaut, welche im Kapitel 13.1 detailliert beschrieben werden.

Die ausgelieferte Konfiguration ermöglicht eine sofortige Validierung von PDF/A, TIFF, SIARD, JP2, JPEG und SIPs.

Nachfolgend werden die Bestandteile kurz beschrieben.

7.1.1 Allgemeiner Teil

xml-Tag	Beschreibung: Initialwert
<pathlogfile>	Pfadangabe zum Logverzeichnis von KOST-Val: logs
<pathtoworkdir>	Pfadangabe zum temporären Arbeitsverzeichnis von KOST-Val: temp_KOST-Val
<pathtodiagnose>	Pfadangabe zum Verzeichnis der Diagnosedaten (siehe auch Kapitel 5): diagnose
<pathtohoveconfig>	Pfadangabe zur mitgelieferten Konfigurationsdatei von Jhove: configuration\jhove.conf

7.1.2 PDF/A Teil

xml-Tag	Beschreibung: Initialwert
<pdfavalidation>	Angabe ob eine PDF/A-Validierung stattfinden soll: yes
<firstvalidator>	Angabe mit welchem Validator in erster Linie die PDF/A-Validierung durchgeführt werden soll: PDFTools
<dualvalidation>	Angabe ob eine duale oder eine einfache Validierung durchgeführt werden soll (siehe auch Kapitel 4): simple

⁸ Die Portable Notepad ++ Applikation kann unter http://portableapps.com/de/apps/development/notepadpp_portable heruntergeladen werden.

xml-Tag	Beschreibung: Initialwert
<pathtopdftronexe>	Pfadangabe zu PDFTron (siehe auch Kapitel 6.2.1): C:\Software\pdfa\pdfa.exe
<pdfa1>	Angabe welche Konformität bei der Version 1 mindestens erreicht werden muss: 1B
<pdfa2>	Angabe welche Konformität bei der Version 2 mindestens erreicht werden muss: 2B

7.1.3 SIARD Teil

xml-Tag	Beschreibung: Initialwert
<siardvalidation>	Angabe ob eine SIARD-Validierung stattfinden soll: yes

7.1.4 JP2 Teil

xml-Tag	Beschreibung: Initialwert
<jp2validation>	Angabe ob eine JP2-Validierung stattfinden soll: yes

7.1.5 JPEG Teil

xml-Tag	Beschreibung: Initialwert
<jpegvalidation>	Angabe ob eine JPEG-Validierung stattfinden soll: yes

7.1.6 TIFF Teil⁹

xml-Tag	Beschreibung: Initialwert
<tiffvalidation>	Angabe ob eine TIFF-Validierung stattfinden soll: yes
<allowedcompressionX>	Angabe ob der Komprimierungsalgorithmus X erlaubt ist oder nicht. Nachfolgende sind im Initialzustand erlaubt: uncompressed, CCITT 1D, CCITT Group 3, CCITT Group 4, LZW, PackBits
<allowedphotointerY>	Angabe ob der Farbraum Y erlaubt ist oder nicht. Nachfolgende sind im Initialzustand erlaubt: white is zero, black is zero, RGB, palette color
<allowedbitspersampleZ>	Angabe ob Z Bits per Sample (pro Kanal) erlaubt sind oder nicht. Nachfolgende sind im Initialzustand erlaubt: 1, 4, 8, 16
<allowedmultipage>	Angabe ob Multipage-TIFFs erlaubt sind oder nicht: 1
<allowedtiles>	Angabe ob der Aufbau in Kacheln erlaubt ist oder nicht: 0

⁹ Die Initialwerte basieren auf der KaD Preservation Planning Empfehlung (http://kost-ceco.ch/cms/index.php?preservation_de), welche unter anderem auf die Baseline-TIFF zurückgreifen. Wo nötig wurden die Baseline-TIFFs präzisiert oder erweitert.

xml-Tag	Beschreibung: Initialwert
<allowedsize>	Angabe ob Dateigrößen von 1'000'000'000 Bytes (~1GB) und grösser erlaubt sind oder nicht: 0

7.1.7 SIP Teil

xml-Tag	Beschreibung: Initialwert
<allowedlengthofpaths>	Erlaubte maximale Anzahl Zeichen in Pfadlängen: 179
<allowedlengthoffiles>	Erlaubte maximale Anzahl Zeichen in Dateinamen: 40
<allowedsipname>	Vorgaben zum Aufbau des SIP-Namens: SIP_[1-2][0-9]{3}[0-1][0-9][0-3][0-9]_lw{3}
<pathtodroidssignature>	Pfadangabe zur verwendeten DROID-Signaturdatei: configuration\KaD_SignatureFile_V71.xml
<allowedversionbar1>	Angabe ob die BAR-SIP Version 1 erlaubt ist: 1
<allowedversionbar4ech1>	Angabe ob die BAR-SIP Version 4 respektive eCH-0160 Version 1 erlaubt ist: 1
<allowedformats>	Auflistung der erlaubten Dateiformate, bestehend aus Unterelementen mit Detailinformationen. Nachfolgende sind im Initialzustand erlaubt: TXT, TIFF, JP2, PDFa1, PDFa2, WAVE, CSV, SIARD

7.2 "KaD_SignatureFile_V71.xml" und "jhove.conf"

Im Ordner "configuration" sind die Dateien "KaD_SignatureFile_V71.xml" und "jhove.conf" abgelegt, welche nicht angepasst werden müssen.

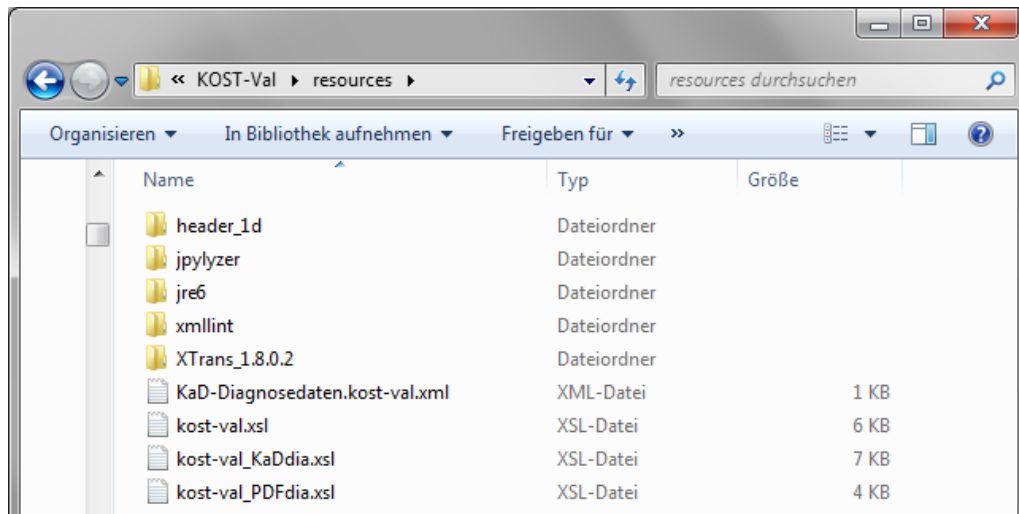
"KaD_SignatureFile_V71.xml" wird für die Formaterkennung benötigt. Es enthält der Stand von DROID per 28.08.2013 und wurde zudem durch die KOST im Bereich der Erkennung der archivtauglichen Formate¹⁰ angepasst.

"jhove.conf" wird für die interne Jhove-Validierung benötigt.

¹⁰ Anstelle des SignatureFile von DROID verwendet KOST-Val das dazu kompatible KaD-SignatureFile der KOST (siehe auch https://github.com/KOST-CECO/KaD_SignatureFile). Dieses umfasst nur die im KaD untersuchten Formate und ermöglicht deren Erkennung in der von der KOST empfohlenen Granularität. Gegenüber dem DROID-SignatureFile werden damit eine signifikante Effizienzsteigerung und eine höhere Benutzerfreundlichkeit erreicht.

8 Ressourcen des KOST-Val

Sämtliche Ressourcen des KOST-Val sind im Unterordner "resources" abgelegt.



9 Validierung starten



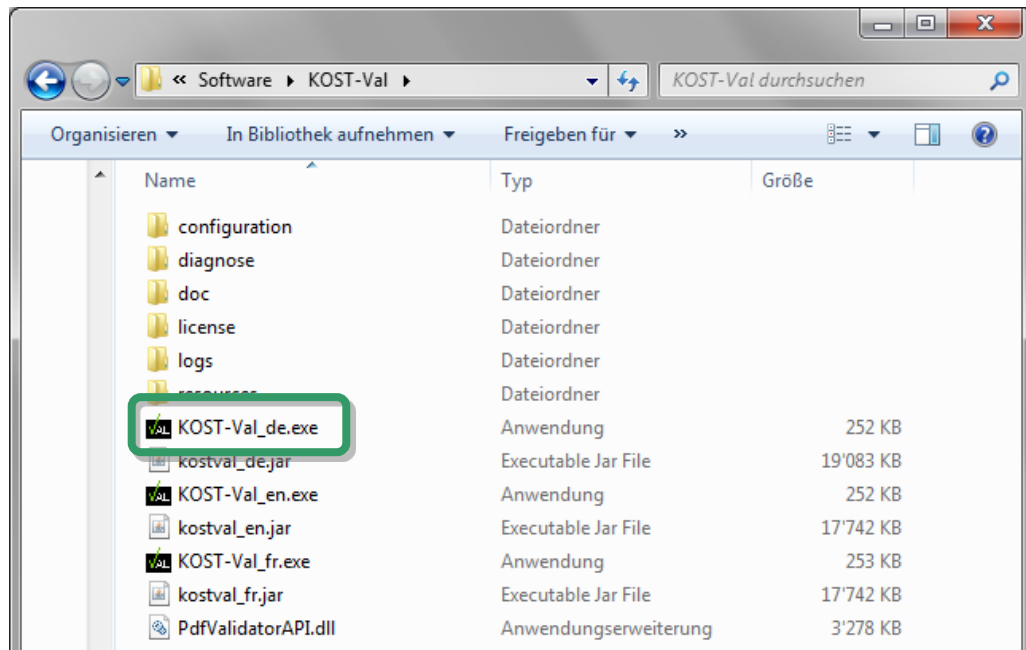
KOST-Val ist nicht Thread-sicher!

Das bedeutet, dass KOST-Val nicht mehrfach gleichzeitig ausgeführt werden kann, ohne sich gegenseitig zu behindern. Wird KOST-Val gleichzeitig ausgeführt, können Fehler wie z.B. eine fehlende Arbeitskopie vorkommen.

9.1 Validierung starten mittels "KOST-Val_de.exe"-GUI

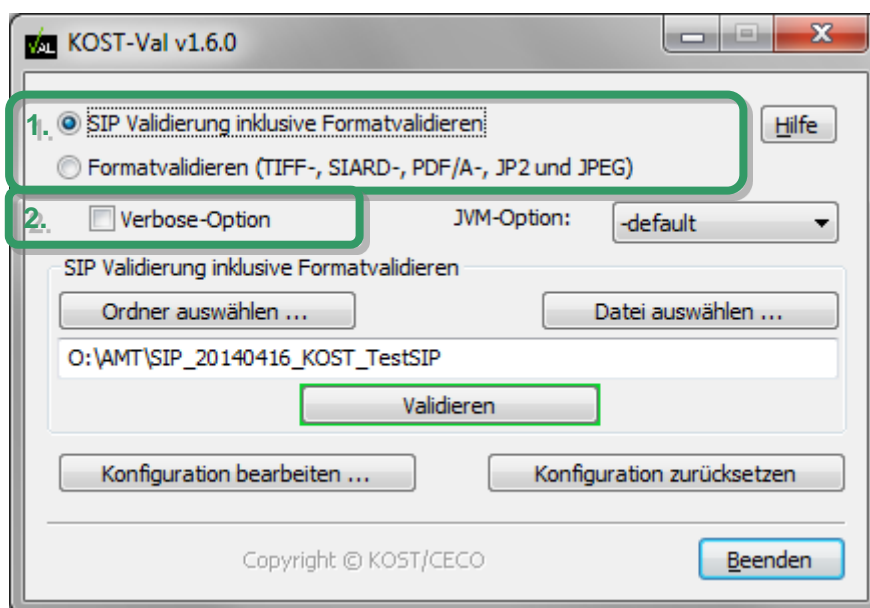
1

Starten von KOST-Val mittels Doppelklick auf "KOST-Val_de.exe" im Ordner "KOST-Val".

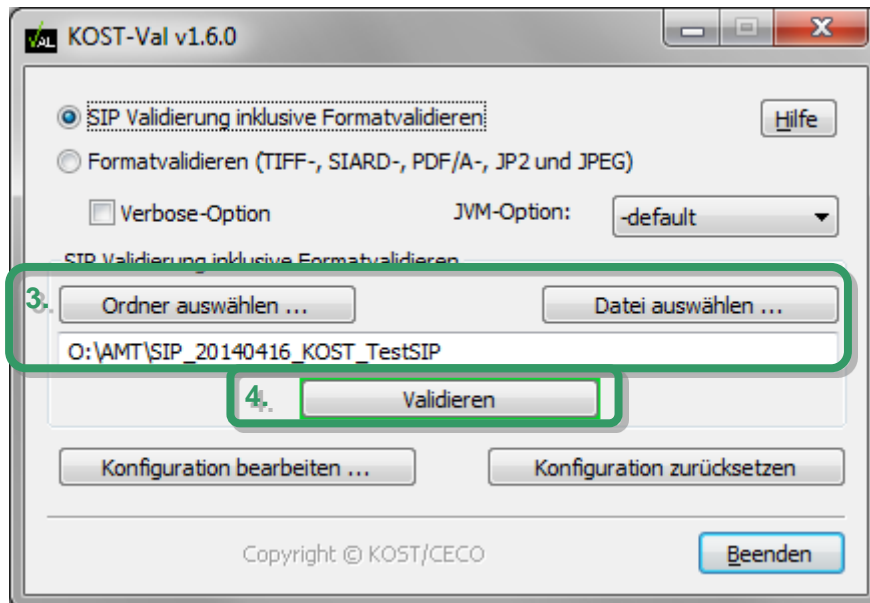


2

1. Auswählen ob eine reine Formatvalidierung oder eine SIP-Validierung inkl. Formatvalidierung durchgeführt werden soll.
2. Auswählen ob Reports von Jhove bzw. Jpylyzer behalten werden sollen oder nicht.



3. Den Pfad zu der zu validierenden Datei oder zur Ordnerstruktur auswählen oder direkt eingeben.
4. Validieren starten

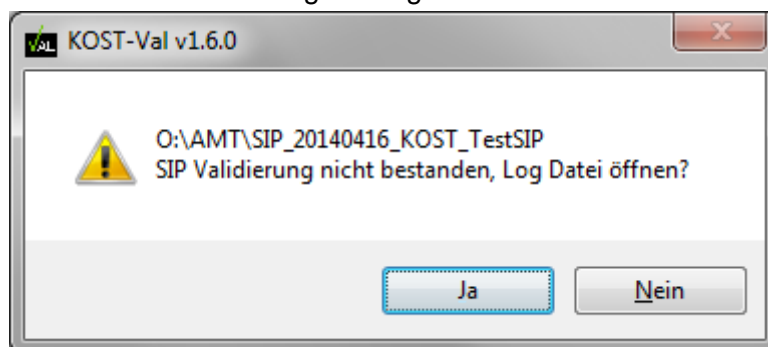


Hinweis:


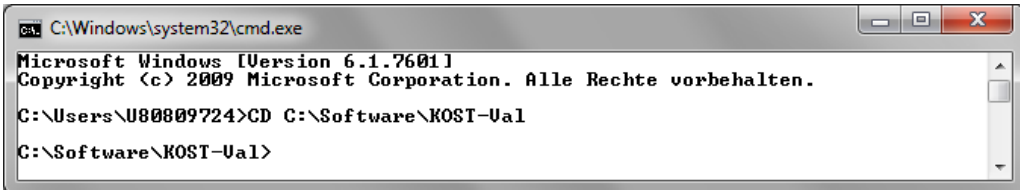
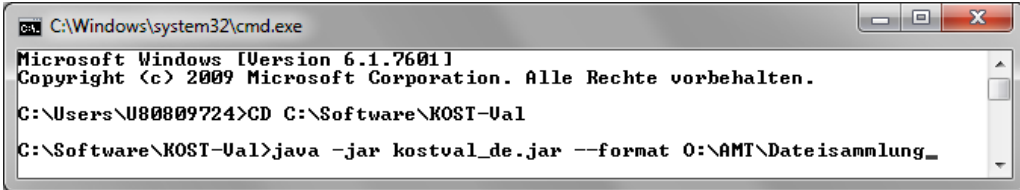
Bei Bedarf kann die Standardeinstellung des virtuellen Java Memory angepasst werden. -Xmx sollte bei „Out of Memory“ und -Xss bei „Stack Overflow“ Fehlern angepasst werden.

Wenn gewünscht kann die Konfiguration temporär angepasst werden. Die temporäre Konfiguration wird beim Betätigen der Schaltflächen "Beenden" und "Konfiguration zurücksetzen" auf den ursprünglichen Stand zurückgesetzt.

- 3 Am Schluss der Validierung wird das Ergebnis angezeigt. Wenn gewünscht kann die KOST-Val-Log-Datei geöffnet werden.



9.2 Validierung manuell starten

1	<p>Der zu validierende Ordner oder Datei bereitstellen.</p> 
2	<p>Eingabeaufforderung öffnen (Start → Ausführen ... → cmd [OK]) und in das gewünschte Arbeitsverzeichnis wechseln (CD C:\Software\KOST-Val)¹¹.</p> 
3	<p>KOST-Val-Programmaufruf starten (die einzelnen Eingabebestandteile mit Leerzeichen trennen).</p>  <p>A resources\jre6\bin\java.exe¹² -jar B kostval_de.jar C --format (oder --sip für die SIP-Validierung) D O:\AMT\Dateisammlung oder O:\AMT\Dateisammlung\jim__ah.tif E "nichts" oder -v (Verbose-Option)</p> <p>A = Java-Befehl (resources\jre6\bin\java.exe -jar), B = relationaler Pfad und die Datei kostval_de.jar, C = --format (oder --sip für die SIP-Validierung), D = relationaler Pfad und Name zum validierenden Ordner oder Datei, E = Verbose-Option, wenn die Originalreports behalten werden sollen.</p> <p><u>Anmerkungen:</u> Wenn ein Eingabebestandteil Leerzeichen enthält, muss dieser in Anführungs- und Schlusszeichen eingegeben werden.</p>

¹¹ Das Laufwerk wird z.B. mit CD /D gewechselt.

¹² Die Eingabe `java -jar` ist nur möglich, wenn Java Runtime Environment (JRE) Version 6 die Standardversion ist. Mit JRE7 kann es zu Abbrüchen kommen. KOST-Val ist im Zusammenhang mit JRE8 massiv langsamer.

Der KOST-Val kann auch von einem beliebigen Ort aus aufgerufen werden. Dies bedingt jedoch die Eingabe von absoluten Pfaden nicht nur im Programmaufruf selber, sondern auch in der Konfigurationsdatei "kostval.conf.xml".



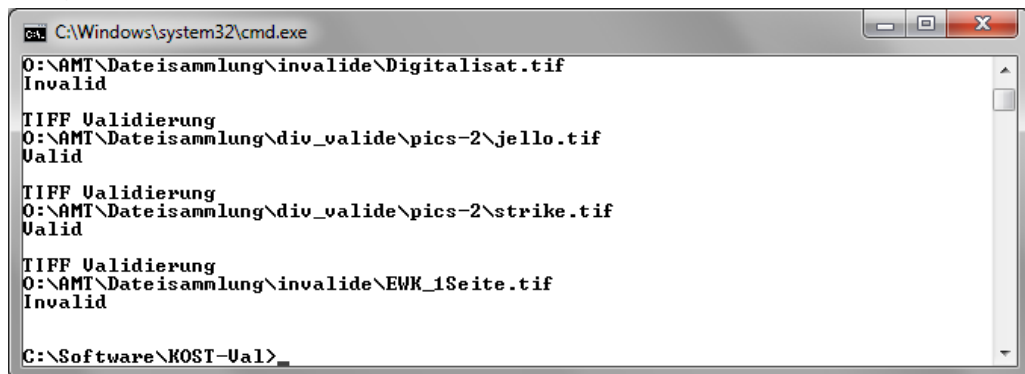
```

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. Alle Rechte vorbehalten.
C:\Users\U80809724>java -jar C:\Software\KOST-Val\kostval_de.jar --format O:\AMT\Dateisammlung

```

- A** C:\Software\KOST-Val\resources\jre6\bin\java.exe -jar
- B** C:\Software\KOST-Val\kostval_de.jar
- C** --format (oder --sip für die SIP-Validierung)
- D** O:\AMT\Dateisammlung oder O:\AMT\Dateisammlung\jim____ah.tif
- E** "nichts" oder -v (Verbose-Option)

- 4** Die Datei wurde validiert, sobald "Valid" oder "Invalid" im cmd-Fenster erscheint. Der Ordner wurde validiert, sobald das Prompt (C:\Software\KOST-Val>) erscheint.



```

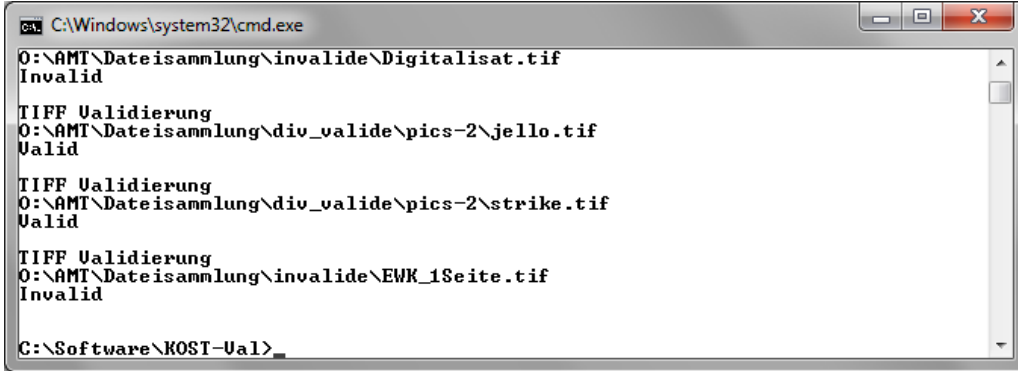
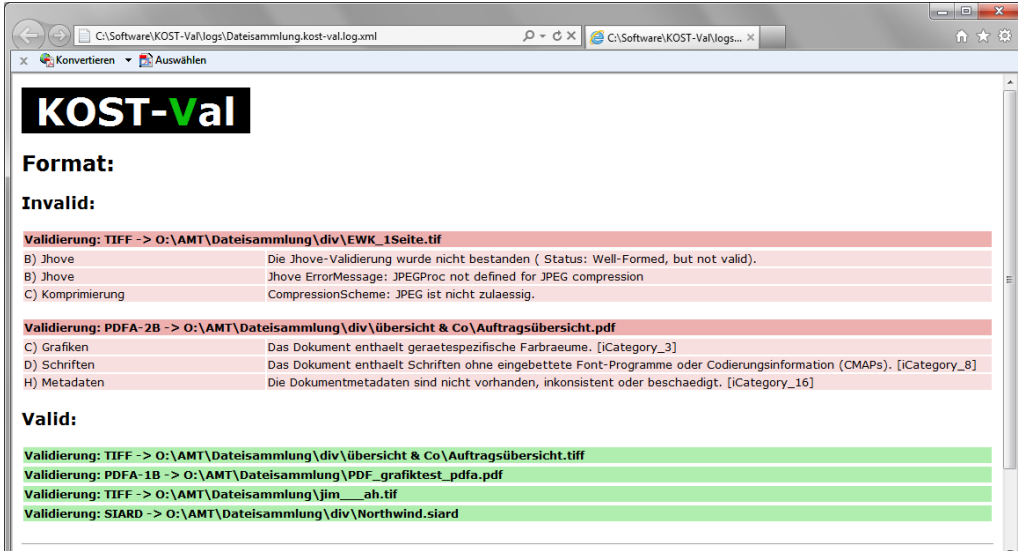
C:\Windows\system32\cmd.exe
O:\AMT\Dateisammlung\invalide\Digitalisat.tif
Invalid
TIFF Validierung
O:\AMT\Dateisammlung\div_valide\pics-2\jello.tif
Valid
TIFF Validierung
O:\AMT\Dateisammlung\div_valide\pics-2\strike.tif
Valid
TIFF Validierung
O:\AMT\Dateisammlung\invalide\EWK_1Seite.tif
Invalid
C:\Software\KOST-Val>

```

Die detaillierten Resultate sind in der kost-val.log.xml-Datei ersichtlich. Das Ergebnis der Gesamtvalidierung (korrekte/fehlerhafte Datei) wird ebenfalls ausgegeben und ist im *exit*-Status des Programms sichtbar, so dass die Validierung in eine automatisierte Verarbeitungskette eingebunden werden kann. Der *exit*-Status kann die folgenden Werte annehmen:

- 0 alles OK
- 1 Fehler im Programmaufruf
- 2 Validierung nicht bestanden

10 Validierung auswerten

1	Die Validierungsschritte sind im Kapitel 11 zusammengestellt.
2	<p>Nach dem Start von KOST-Val werden die Validierungsart und die Datei angezeigt. Wenn der Validierung ohne Fehler abgeschlossen wurde, wird "Valid" ausgegeben.</p> <p>Sollte eine Validierung fehlerhaft sein, wird "Invalid" ausgegeben.</p>  <p>Wurde die Option "-v" (Verbose für "gesprächig") eingegeben, wird bei der Formatvalidierung zusätzlich auch der Originalreport ins Logverzeichnis geschrieben. Die Option "-v" ist nur für Formatspezialisten in seltenen Fällen interessant, da alle Fehlermeldungen 1:1 von KOST-Val übernommen werden.</p>
3	<p>Die log-Datei führt zusätzliche Detailinformationen über die einzelnen invaliden Validierungsschritte auf, insbesondere der betroffene Validierungsschritt und der entsprechende Fehler.</p> 

11 Beschreibung der Validierungsschritte

11.1 TIFF

Nachfolgend werden die einzelnen TIFF-Validierungsschritte detaillierter beschrieben.

A Erkennung

Wird die TIFF-Datei als TIFF erkannt?

B Jhove

Wurde die Jhove-Validierung bestanden?

C Komprimierung

Ist die verwendete Komprimierung gemäss der Konfiguration erlaubt?

D Farbraum

Ist der verwendete Farbraum gemäss der Konfiguration erlaubt?

E BitsPerSample

Sind die verwendeten BitsPerSample gemäss der Konfiguration erlaubt?

F Multipage

Entspricht die Verwendung der Eigenschaft „Multipage“ der Konfiguration?

G Kacheln

Entspricht die Verwendung der Eigenschaft „Kacheln“ der Konfiguration?

H Grösse

Ist die Dateigrösse von der Konfiguration erlaubt?

11.2 SIARD

Nachfolgend werden die einzelnen Validierungsschritte detaillierter beschrieben. Diese Kriterien sind ein Auszug der Muss-Kriterien aus der SIARD-Spezifikation.

- A Lesbarkeit**
Kann die SIARD-Datei gelesen werden?
- B primäre Verzeichnisstruktur**
Besteht eine korrekte primäre Verzeichnisstruktur?
- C Header-Validierung**
Ist der header-Ordner valid?
- D Struktur-Validierung**
Stimmt die Struktur aus metadata.xml mit der Datei-Struktur von content überein?
- E Spalten-Validierung**
Wurden die Angaben aus metadata.xml korrekt in die tableZ.xsd-Dateien übertragen?
- F Zeilen-Validierung**
Wurden die Angaben aus metadata.xml korrekt in die tableZ.xsd-Dateien übertragen?
- G Tabellen-Validierung**
Sind die Spaltennamen innerhalb der Tabelle einmalig?
- H Content-Validierung**
Sind die XML-Dateien im content valid zu ihrer Schema-Definition (XSD-Dateien)?
- I SIARD-Erkennung**
Wird die SIARD-Datei als SIARD erkannt?
- J Zusätzliche Primärdateien**
Sind alle im content-Ordner enthaltenen Dateien oder Ordner in metadata.xml beschrieben?

11.3 PDF/A

Die einzelnen Meldungen im Originalreport von PDFTron werden anhand der ersten Zahl des "Error ID" den Modulen zugeordnet. Die Fehlermeldungen (Message) werden 1:1 übernommen, sind jedoch übersetzt (Original in Englisch). Bei der Validierung mit PDF-Tools werden die Kategorien den Modulen zugeordnet und der Text ausgegeben.

A Allgemeines

Sind die allgemeinen Anforderungen wie z.B. nicht korrupte PDF/A-Datei erfüllt?

B Struktur

Ist die PDF/A-Datei korrekt aufgebaut (z.B. BOF und EOF)?

C Grafiken

Sind die Grafiken korrekt in der PDF/A-Datei integriert (z.B. valider Farbraum)?

D Schriften

Sind die Schriften korrekt in der PDF/A-Datei eingebettet (z.B. sind alle verwendeten Zeichen eingebettet)?

E Transparenz

Sind die Anforderungen betreffend Transparenz an die PDF/A-Datei erfüllt?

F Annotationen & Interaktionen

Enthält die PDF/A-Datei nur erlaubte Annotationen (z.B. keine 3D-Annotation)?
Enthält die PDF/A-Datei nur erlaubte Interaktionen?

G Aktionen

Enthält die PDF/A-Datei nur erlaubte Aktionen (z.B. kein JavaScript)?

H Metadaten

Sind die Metadaten korrekt in der PDF/A-Datei enthalten?

I Zugänglichkeit

Sind die Anforderungen an die Zugänglichkeit (Conformance Level A) erfüllt?

11.4 JP2

Die nicht bestandenen Tests von Jpylyzer werden den Modulen zugeordnet, und entsprechende Fehlermeldungen werden ausgegeben. Nachfolgend werden die einzelnen JP2-Validierungsschritte detaillierter beschrieben.

A Erkennung und Jpylyzer

Wird die JP2-Datei als JP2 erkannt? Wurde die Jpylyzer-Validierung bestanden?

B Metadaten-Validierung

Wurden die Metadaten korrekt erfasst?

C Bild-Validierung

Ist das Bild korrekt aufgebaut?

D Erweiterte Validierung

Wurden die zusätzlichen Jpylyzer-Tests bestanden?

11.5 JPEG

Die Fehlermeldungen aus BadPeggy werden den Modulen zugeordnet, und übersetzt in Deutsch ausgegeben (Original in Englisch). Nachfolgend werden die einzelnen JPEG-Validierungsschritte detaillierter beschrieben.

A Erkennung und BadPeggy

Wird die JPEG-Datei als JPEG erkannt? Wurde die BadPeggy-Validierung bestanden?

B Korrupte Daten

Gibt BadPeggy eine Fehlermeldung heraus, welche auf korrupte Daten hinweist?

C Ungültige Dateistruktur

Gibt BadPeggy eine Fehlermeldung heraus, welche auf eine ungültige Dateistruktur hinweist?

D Andere Probleme

Gibt BadPeggy eine Fehlermeldung aus, welche noch nicht den Modulen zugeordnet und übersetzt wurde?

11.6 SIP

Nachfolgend werden die einzelnen Validierungsschritte detaillierter beschrieben. Diese Kriterien sind ein Auszug der Muss-Kriterien aus der SIP-Spezifikation.

Modul 1: Paket- und XML-Konsistenz

1A Lesbarkeit

Kann das SIP (ZIP / ZIP64) fehlerfrei geöffnet werden?

1B primäre Verzeichnisstruktur

Besteht eine korrekte primäre Verzeichnisstruktur?

1C Verzeichnis- und Dateinamen

Entsprechen die Namen den Einschränkungen in der Spezifikation?

1D Schemavalidierung metadata.xml

Entspricht metadata.xml den Schemadateien in /header/xsd?

1E SIP-Typ ermitteln

Der SIP Typ wird ermittelt und angezeigt: GEVER oder FILE

1F Primärdateien im Verzeichnis

Sind Primärdateien im Verzeichnis /content vorhanden?¹³

Modul 2: Datei-Konsistenz

2A Fehlende Primärdateien

Sind alle referenzierten Dateien vorhanden?

2B Zusätzliche Primärdateien

Sind im SIP keine zusätzlichen Primärdateien vorhanden?

2C Prüfsummen-Validierung

Stimmen die Prüfsummen der Dateien mit Prüfsumme überein?

2D Verzeichnung GEVER-Primärdateien

Sind alle referenzierten Dateien auch im Ordnungssystem verzeichnet?

Modul 3: Dateiformat- und Datums-Konsistenz

3A Formaterkennung

Sind die erkannten Formate erlaubt?

3B Zusätzliche Formate

Alle Dateien in nicht erlaubten Formaten mit entsprechenden Formatangaben auflisten.

3C Formatvalidierung

Sind die Formate valid?

3D Zeitraum-Validierung

Stimmen die Zeitangaben in (metadata.xml)/ablieferung überein?

¹³ Primärdateien können in einem GEVER-SIP fehlen, das nur zur Archivierung einer Ordnerstruktur dient, aber nicht in einem FILE-SIP.

12 Urheberrecht

KOST-Val ist eine Entwicklung der KOST. Alle Rechte liegen bei der KOST. KOST-Val wurde im 2012 durch die KOST unter der GNU General Public License v3+ veröffentlicht.

Notice:	This product includes software developed by the Apache Software Foundation (http://www.apache.org/).
----------------	---

KOST-Val stützt sich auf folgende unveränderte Komponenten anderer Hersteller, welche direkt im Quellcode von KOST-Val eingebunden sind:

Drittprogramm / -Komponente	Version	Lizenz
3-Heights™ PDF/A Validator API http://www.pdf-tools.com		Siehe Kapitel 12.2
Apache Commons http://commons.apache.org/ - commons-collections-3.2.1.jar - commons-configuration-1.6.jar - commons-digester-1.8.jar - commons-lang-2.4.jar - commons-logging-1.1.1.jar	3.2.1 1.6 1.8 2.4 1.1.1	Apache License 2.0
Apache log4j http://logging.apache.org/log4j/	1.2.12	Apache License 2.0
Apache Xalan-Java http://xml.apache.org/xalan-j/	2.7.0	Apache License 2.0
Apache Xerces http://xerces.apache.org/	2.7.1	Apache License 2.0
BadPeggy http://coderslagoon.com/	2.0	GPL v3 License
DROID http://sourceforge.net/apps/mediawiki/droid/	5.0.3	3c BSD- License
Jdom 2.0.0 http://www.jdom.org/	2.0.0	jdom License
Jhove http://hul.harvard.edu/jhove/	1.5	LGPL v2.1 License
Junit 4.4 http://www.junit.org/	4.4	CPL v1.0
Spring Framework API http://static.springsource.org/spring/docs/3.0.x/api/	3.0.0	Apache License 2.0
zip64 http://sourceforge.net/projects/zip64file/	1.02	GPL v2+ License

KOST-Val GUI verwendet folgende Komponenten anderer Hersteller:

Drittprogramm / -Komponente	Version	Lizenz
Jpylyzer http://jpylyzer.openpreservation.org/	1.13.0	LGPL v3.0 License
NSIS v2.46 http://nsis.sourceforge.net/Main_Page	2.46	zlib/libpng License
XML.nsh http://nsis.sourceforge.net/XML_plugin	2.0	zlib/libpng License
Xmllint http://xmlsoft.org/xmllint.html/	20630	MIT License
XTrans http://sourceforge.net/projects/xtrans/	1.8.0.2	GPL v2 License

Die Benutzer von KOST-Val sind gehalten, die Lizenzbestimmungen all dieser Komponenten zu befolgen, welche im Verzeichnis KOST-Val\license vorliegen.

12.1 PDF/A Manager von PDFTron

Für die Durchführung einer PDF/A-Validierung wurde auch die indirekte Verwendung (via Konsolenaufwurf) die kommerzielle Software PDF/A Manager von PDFTron implementiert. Diese Software ist nicht Bestandteil von KOST-Val und muss zur allfälligen Verwendung separat beschafft und installiert werden. Weitere Informationen dazu siehe Kapitel 6.2.1.

12.2 3-Heights™ PDF/A Validator API-Lizenz

Für die Verwendung der Eingeschränkten Version des 3-Heights™ PDF/A Validator von PDF-Tools hat die KOST folgende Lizenz mit PDF-Tools vereinbart:

Anhang

zu den Allgemeinen Lizenzbedingungen der PDF Tools AG vom 01.03.2012, Version 2.3

zwischen

PDF Tools AG
Geerenstrasse 33
CH-8185 Winkel

und

KOST
c/o Schweizerisches Bundesarchiv
Archivstrasse 24
CH-3003 Bern

(nachfolgend "**PDF TOOLS**" genannt)

(nachfolgend "**KOST**" genannt)

1. Vorbemerkung

Die untenstehenden Bedingungen sind Ergänzungen der Allgemeinen Lizenzbedingungen der PDF TOOLS und werden mit Unterzeichnung dieser Vereinbarung integrierter Vertragsbestandteil.

2. Individuelle Vereinbarung

Dieses Vertragsverhältnis regelt die Client-Lizenz zwischen der PDF TOOLS als Lizenzgeber und der KOST als Lizenznehmer gemäss nachfolgenden Spezialbestimmungen:


- PDF Tools AG erteilt für KOST eine kostenfreie OEM-Lizenz für das 3-Heights™ PDF/A Validator API als Zusatzfunktion ihrer eigenen Validator-Software (KOST-Val).
- Die Lizenz schliesst den Gebrauch der Software (KOST-Val) durch Gedächtnisinstitutionen, bestehend aus Archiven oder Bibliotheken, deren Zulieferer und der KOST selbst, ein.
- Der OEM-Lizenzschlüssel, welcher fest in KOST-Val eingebunden ist, darf nicht ausserhalb der Applikation (KOST-Val) verwendet werden.
- Die Lizenz ist zeitlich unbegrenzt, jedoch bezüglich Durchsatz pro Installation begrenzt (72'000 Seiten pro Jahr).
- Für die Verteilung der Software (KOST-Val) an den Anwender ist die KOST zuständig.
- Der First Level Support der Anwender erfolgt durch KOST. Second Level Support Fälle leitet KOST an PDF Tools AG weiter.
- Wenn der Anwender weitergehende Bedürfnisse hat, z.B. höherer Durchsatz, Integration in andere Applikationen etc. kauft er die Software (3-Heights™ PDF/A Validator API) direkt bei PDF Tools AG.
- Die KOST darf weiterhin den Quellcode von KOST-Val Open Source publizieren und KOST-Val gratis und ohne Registrierung abgeben.

3. Kundenreferenz

PDF TOOLS ist berechtigt, KOST als Kundenreferenz aufzuführen und die Zusammenarbeit und die Anwendung des Produktes der PDF TOOLS bei KOST zu beschreiben und zu publizieren.

Winkel, 7.10.2014

PDF Tools AG


Dr. Hans Bäruss

Bern, 29. Sept. 2014

KOST


Für den Steuerungsausschuss
Dr. Beat Gnädinger

Für die Benutzer sind folgende Punkte massgebend:

- Die Lizenz schliesst den Gebrauch der Software (KOST-Val) durch Gedächtnisinstitutionen, bestehend aus Archiven oder Bibliotheken, deren Zulieferer und der KOST selbst, ein.
- Der OEM-Lizenzschlüssel, welcher fest in KOST-Val eingebunden ist, darf nicht ausserhalb der Applikation (KOST-Val) verwendet werden.
- Die Lizenz ist zeitlich unbegrenzt, jedoch bezüglich Durchsatz pro Installation begrenzt (72'000 Seiten pro Jahr).
- Der First Level Support der Anwender erfolgt durch KOST. Second Level Support Fälle leitet KOST an PDF Tools AG weiter.
- Wenn der Anwender weitergehende Bedürfnisse hat, z.B. höherer Durchsatz, Integration in andere Applikationen etc. kauft er die Software (3-Heights™ PDF/A Validator API) direkt bei PDF Tools AG.

Die Benutzer von KOST-Val sind gehalten, diese Lizenzbestimmung zu befolgen.

13 Anhang

13.1 Beschreibung der Konfigurationsdatei "kostval.conf.xml"

Die Konfigurationsdatei "kostval.conf.xml" ist in verschiedenen Teilen aufgebaut, welche nachfolgend detailliert beschrieben werden.

13.1.1 Allgemeiner Teil

<pathlogfile>

Pfadangabe zum Logverzeichnis von KOST-Val. Der Initialwert ist **logs**. Dabei kann der absolute Pfad angegeben werden oder wie im Initialwert der relative Pfad (aus Sicht von kostval_de.jar). Dieser Ordner muss nicht angelegt werden.

<pathtoworkdir>

Pfadangabe zum temporären Arbeitsverzeichnis von KOST-Val. Der Initialwert ist **temp_KOST-Val**. Dabei kann der absolute Pfad angegeben werden oder wie im Initialwert der relative Pfad (aus Sicht von kostval_de.jar). Dieser Ordner muss nicht angelegt werden.

<pathtodiagnostics>

Pfadangabe zum Verzeichnis der Diagnosedaten (siehe auch Kapitel 5). Der Initialwert ist **diagnostics**. Dabei kann der absolute Pfad angegeben werden oder wie im Initialwert der relative Pfad (aus Sicht von kostval_de.jar). Dieser Ordner muss nicht angelegt werden.

<pathtojhoconfig>

Pfadangabe zur Konfigurationsdatei von JHOVE. Der Initialwert ist **configuration\jho.conf**. Dabei kann der absolute Pfad angegeben werden oder wie im Initialwert der relative Pfad (aus Sicht von kostval_de.jar).

13.1.2 PDF/A-Teil

<pdfavalidation>

Angabe, ob eine PDF/A-Validierung stattfinden soll (yes) oder nicht (no). Der Initialwert ist **yes**.

<firstvalidator>

Angabe, mit welchem Validator in erster Linie die PDF/A-Validierung durchgeführt werden soll. Implementiert sind folgende PDF/A-Validatoren: 3-Heights™ PDF/A Validator von PDF-Tools (PDFTools) und PDF/A Manager von PDFTron (PDFTron). Der Initialwert ist **PDFTools**.

<dualvalidation>

Angabe, ob eine duale (dual) oder eine einfache (simple) Validierung durchgeführt werden soll (siehe auch Kapitel 4). Der Initialwert ist **simple**.

<pathtopdftronexe>

Pfadangabe zu PDFTron. Der Initialwert ist **C:\Software\pdfa\pdfa.exe**. Dabei kann der absolute Pfad angegeben werden oder der relative Pfad (aus Sicht von kostval_de.jar).

<pdfa1>

Angabe, welche Konformität in der Version 1 mindestens erreicht werden muss. Dabei kann gewählt werden zwischen *Accessible* (1A), *Basic* (1B) oder nicht erlaubt (no). Der Initialwert ist **1B**.

<pdfa2>

Angabe, welche Konformität in der Version 2 mindestens erreicht werden muss. Dabei kann gewählt werden zwischen *Accessible* (2A), *Basic* (2B), *Unicode* (2U) oder nicht erlaubt (no). Der Initialwert ist **2B**.

13.1.3 SIARD-Teil

<siardvalidation>

Angabe, ob eine SIARD-Validierung stattfinden soll (yes) oder nicht (no). Der Initialwert ist **yes**.

13.1.4 JP2-Teil

<jp2validation>

Angabe, ob eine JP2-Validierung stattfinden soll (yes) oder nicht (no). Der Initialwert ist **yes**.

13.1.5 JPEG-Teil

<jpegvalidation>

Angabe, ob eine JPEG-Validierung stattfinden soll (yes) oder nicht (no). Der Initialwert ist **yes**.

13.1.6 TIFF-Teil

Die Initialwerte basieren auf der KaD Preservation Planning Empfehlung (http://kost-ceco.ch/cms/index.php?preservation_de), welche unter anderem auf Baseline TIFF zurückgreifen. Wo nötig wurde Baseline TIFF präzisiert oder erweitert.

<tiffvalidation>

Angabe, ob eine TIFF-Validierung stattfinden soll (yes) oder nicht (no). Der Initialwert ist **yes**.

<allowedcompressionX>

Angabe, ob der Komprimierungsalgorithmus erlaubt ist (Beschreibung) oder nicht (0). Die Initial erlaubten Komprimierungsalgorithmen sind **uncompressed, CCITT 1D, CCITT Group 3, CCITT Group 4, LZW, PackBits**.

X	Beschreibung	Initialwert	Herkunft
1	uncompressed	uncompressed	Baseline-TIFF
2	CCITT 1D	CCITT 1D	Baseline-TIFF
3	CCITT Group 3	CCITT Group 3	Baseline-TIFF
4	CCITT Group 4	CCITT Group 4	Baseline-TIFF
5	LZW	LZW	KaD-PP Erweiterung
7	ISO JPEG	0	Baseline-TIFF
8	Deflate	0	Baseline-TIFF
32773	PackBits	PackBits	Baseline-TIFF

<allowedphotointerY>

Angabe, ob der Farbraum erlaubt ist (Beschreibung) oder nicht (0). Die Initial erlaubten Farbräume sind **white is zero, black is zero, RGB, palette color**.

Y	Beschreibung	Initialwert	Herkunft
0	white is zero	white is zero	Baseline-TIFF
1	black is zero	black is zero	Baseline-TIFF
2	RGB	RGB	Baseline-TIFF
3	palette color	palette color	Baseline-TIFF
4	transparency mask	0	Baseline-TIFF
5	CMYK	0	Baseline-TIFF
6	YCbCr	0	Baseline-TIFF
7	CIE L*a*b*	0	Baseline-TIFF

<allowedbitspersampleZ>

Angabe, ob die Bits per Sample (pro Kanal) erlaubt sind (Z) oder nicht (0). Die Initial erlaubten Bits per Sample sind **1, 4, 8, 16**.

Z	Beschreibung	Initialwert	Herkunft
1	1 Bits per Sample (pro Kanal)	1	Baseline-TIFF
2	2 Bits per Sample (pro Kanal)	0	Baseline-TIFF
4	4 Bits per Sample (pro Kanal)	4	Baseline-TIFF
8	8 Bits per Sample (pro Kanal)	8	Baseline-TIFF
16	16 Bits per Sample (pro Kanal)	16	KaD-PP Erweiterung
32	32 Bits per Sample (pro Kanal)	0	Baseline-TIFF
64	64 Bits per Sample (pro Kanal)	0	Baseline-TIFF

<allowedmultipage>

Angabe, ob Multipage-TIFFs erlaubt sind (1) oder nicht (0). Der Initialwert ist **1** und entspricht der KaD Preservation Planning Empfehlung TIFF.

<allowedtiles>

Angabe, ob der Aufbau in Kacheln erlaubt ist (1) oder nicht (0). Der Initialwert ist **0** und entspricht den Vorgaben der Baseline-TIFFs.

<allowedsize>

Angabe, ob Dateigrößen von 1'000'000'000 Bytes (~1GB) und grösser erlaubt sind (1) oder nicht (0). Der Initialwert ist **0** und entspricht der KaD Preservation Planning Empfehlung TIFF.

13.1.7 SIP Teil

<allowedlengthofpaths>

Erlaubte maximale Anzahl Zeichen in Pfadlängen. Der Initialwert ist **179** und entspricht der Einschränkung in der SIP-Spezifikation: weniger als 180 Zeichen.

<allowedlengthoffiles>

Erlaubte maximale Anzahl Zeichen in Dateinamen. Der Initialwert ist **40** und entspricht der Einschränkung im BAR-SIP (Version 1): maximal 40 Zeichen.

Bei der Validierung der neueren SIP-Versionen gibt es keine Einschränkungen, entsprechend ist dieses Feld nur für die BAR-SIP Version 1 relevant.

<allowsipname>

Vorgaben zum Aufbau des SIP-Namens. Der Initialwert (Regex-Ausdruck) ist **SIP_[1-2][0-9]{3}[0-1][0-9][0-3][0-9]_\\w{3}** und entspricht der Einschränkung i in der SIP-Spezifikation: SIP_[Ablieferungsdatum]_[Name der Abliefernden Stelle](_[Referenz]), wobei das Datum in der Form JJJJMMTT angegeben werden muss und die Referenz optional verwendet werden kann.

<pathtodroidsignature>

Pfadangabe zur verwendeten DROID konformen Signaturdatei im Verzeichnis. Der Initialwert ist **configuration\\KaD_SignatureFile_V71.xml**. Weitere Informationen sind im Kapitel 7.2 ersichtlich.

<allowedversionbar1>

Angabe ob die BAR-SIP-Version 1 erlaubt (1) ist oder nicht (0). Der Initialwert ist **1**.

<allowedversionbar4ech1>

Angabe ob die BAR-SIP-Version 4 respektive eCH-0160 Version 1 erlaubt (1) ist oder nicht (0). Der Initialwert ist **1**.

<allowedformats>

Auflistung der erlaubten Dateiformate bestehend aus Unterelementen mit Detailinformationen. Die Initialwerte sind aus der nachfolgenden Tabelle ersichtlich. Die Liste ist bewusst ausführlich gehalten, so dass die nicht erlaubten Formate einfach entfernt werden können und möglichst wenig nachgetragen werden muss. Angaben betreffend der Extension und der PUID sind der KaD_SignatureFile_V70.01.xml zu entnehmen.

<puid> </puid>	<extension> </extension>	<comment> </comment>
TXT	txt	Plain Text file
TIFF	tif, tiff	Tagged Image File Format
JP2	jp2, jpg2, j2c	JPEG2000
PDFA1	pdf	PDF/A-1
PDFA2	pdf	PDF/A-2
WAVE	wav, wave	Waveform Audio
CSV	csv	Comma Separated Values
SIARD	siard	SIARD
XML	xml	Ext. Markup Language
XSD	xsd	XML Schema Definition
JPEG	jpeg, jpg, jpe	JPEG
MP3	mp3	MPEG 1/2 Audio Layer 3
MP4	mp4, m4v, m4a, f4v, f4a	MPEG-4 Media File

13.2 Programmaufbau

KOST-Val wurde nach folgenden Anforderungen aufgebaut:

Funktionale Anforderungen:

TIFF-Validierung: KOST-Val liest ein TIFF und validiert mit Hilfe von Jhove die folgenden Punkte:

Validierungsschritt	Bezeichnung
A (Abbruch wenn Fehler)	Erkennung
B	Jhove
C	Komprimierung
D	Farbraum
E	BitsPerSample
F	Multipage
G	Kacheln
H	Grösse

SIARD-Validierung: KOST-Val liest ein SIARD und validiert die folgenden Punkte:

Validierungsschritt	Bezeichnung
A (Abbruch wenn Fehler)	Lesbarkeit
B (Abbruch wenn Fehler)	Primäre Verzeichnisstruktur
C (Abbruch wenn Fehler)	Header-Validierung
D (Abbruch wenn Fehler)	Struktur-Validierung
E	Spalten-Validierung
F	Zeilen-Validierung
G	Tabellen-Validierung
H	Content-Validierung
I	SIARD-Erkennung
J	Zusätzliche Primärdateien

PDF/A-Validierung: KOST-Val liest ein PDF und validiert mit Hilfe des 3-Heights™ PDF/A Validator von PDF-Tools oder auch des PDF/A-Manager von PDFTron die folgenden Punkte:

Validierungsschritt	Bezeichnung
A (Abbruch wenn Fehler)	Allgemeines
B	Struktur
C	Grafiken
D	Schriften
E	Transparenz
F	Annotationen
G	Aktionen & Interaktionen
H	Metadaten
I	Zugänglichkeit

JP2-Validierung: KOST-Val liest ein JP2 und validiert mit Hilfe von Jpylyzer die folgenden Punkte:

Validierungsschritt	Bezeichnung
A (Abbruch möglich wenn Fehler)	Erkennung und Jpylyzer
B	Metadaten-Validierung
C	Bild-Validierung
D	Erweiterte Validierung

JPEG-Validierung: KOST-Val liest ein JPEG und validiert mit Hilfe von BadPeggy die folgenden Punkte:

Validierungsschritt	Bezeichnung
A (Abbruch möglich wenn Fehler)	Erkennung und BadPeggy
B	Korrupte Daten
C	Ungültige Dateistruktur
D	Andere Probleme

SIP-Validierung: KOST-Val liest ein SIP und validiert die folgenden Punkte aus der SIP Spezifikation:

Validierungsschritt	Bezeichnung (Stepname)
1a (Abbruch wenn Fehler)	Lesbarkeit
1b (Abbruch wenn Fehler)	primäre Verzeichnisstruktur
1c (Abbruch wenn Fehler)	Verzeichnis- und Dateinamen
1d (Abbruch wenn Fehler)	Schemavalidierung metadata.xml
1e	SIP-Typ ermitteln
1f	Primärdateien im Verzeichnis
2a	Fehlende Primärdateien
2b	Zusätzliche Primärdateien
2c	Prüfsummen-Validierung
2d	Verzeichnung GEVER-Primärdateien
3a	Formaterkennung
3b	Zusätzliche Formate
3c	Formatvalidierung
3d	Zeitraum-Validierung

Die Resultate (inklusive Meldungen zu Inkonsistenzen oder Fehler) werden pro Schritt ausgegeben und in eine Validierungs-Logdatei geschrieben.

Das Ergebnis der Gesamtvalidierung (korrekte/fehlerhafte Datei) wird ebenfalls ausgegeben und im *exit*-Status des Programms sichtbar, so dass die Validierung in eine automatisierte Verarbeitungskette eingebunden werden kann. Der *exit*-Status kann die folgenden Werte annehmen:

- 0 alles OK
- 1 Fehler im Programmaufruf
- 2 Validierung nicht bestanden

Die einzelnen Validierungsschritte / Prüfungen werden nacheinander ausgeführt. Wo möglich, wird die Validierung auch bei Fehlern weiter fortgesetzt, um die Anzahl von Korrekturzyklen zu reduzieren.

Nichtfunktionale Anforderungen:

Für besondere Aufgaben werden externe Programme oder entsprechende Java-Frameworks eingesetzt.

Die Anwendung ist modular aufgebaut, damit ohne viel Aufwand ein oder mehrere weitere Validierungsmodule eingebaut werden können.

Die Log-/Programmausgabe erlaubt ein einfaches Auslesen des Ergebnisses der einzelnen Validierung und damit die Verwendung des Tools in einer Prozesskette,

Die Konsolenausgabe begrenzt sich auf die Bezeichnung der Validierungsart, das Gesamtergebnis "valid" oder "invalid" sowie der Pfad zur Datei. Alle zusätzlichen Informationen werden in der Log-Datei aufgeführt.