



Abridged Manual

[translation in progress]

Content

1	Preface	2
2	System requirements	3
3	Open issues / Feedback	3
4	Einleitung duale PDF/A-Validierung [translation in progress]	4
5	Diagnosedaten [translation in progress]	4
6	Installation	5
6.1	Installation of KOST-Val	5
6.2	Optional installation of third-party applications	5
7	Configuration of KOST-Val	6
7.1	Parts of the configuration file "kostval.conf.xml"	6
7.2	"KaD_SignatureFile_V71.xml" and "jhove.conf"	8
8	Resources of KOST-Val	8
9	Start the validation	9
9.1	Start the validation using the "KOST-Val_en.exe" GUI	9
9.2	Start the validation manually	11
10	Interpret the validation results	13
11	Copyright	14
11.1	3-Heights™ PDF/A Validator API-Lizenz [translation in progress]	15
11.2	PDF/A Manager by PDFTron	16
12	Annex	17
12.1	Program structure	17
12.2	Functional Principle of Format Validation	19



This Manual is only an abridged Manual.
Check the German or French Manual for more Information.

1 Preface

KOST-Val is a Java-based application for validating the structure and content of TIFF (*Tagged Image File Format*), SIARD (*Software Independent Archiving of Relational Databases*), PDF/A, and JP2 (JPEG 2000; *Joint Photographic Experts Group 2000*) files as well as Submission Information Packages (SIP) for digital information ingest. It is an open source application under a GPL v3+ licence. KOST-Val uses unmodified components of other manufacturers by embedding them directly into the source code. Users of KOST-Val are requested to adhere to these components' terms of licence. Please refer to chapter 11 for further information.

KOST-Val complies with the following requirements.

TIFF validation: KOST-Val reads a TIFF file and uses JHOVE to validate the structure, the content, and key properties such as compression, colour space, and multipage. These properties can be configured.

SIARD validation: KOST-Val reads a SIARD (eCH-0165 v1¹) file and validates the structure and the content.

PDF/A validation: KOST-Val reads a PDF or PDF/A file (ISO 19005-1 and 19005-2) and uses 3-Heights™ PDF/A Validator by PDF-Tools or PDF/A Manager by PDFTron to validate the structure and the content of the PDF file. KOST-Val organises the different error messages into main categories such as fonts, graphics, and metadata. KOST-Val supplies only a limited² version from 3-Heights™ PDF/A Validator by PDF-Tools.

Module J extracts and validates the JPEG and JP2 images contained in the PDF file (depending on the configuration)³. It is also possible to configure whether the JBIG2 compression⁴ is accepted or not.

JP2 validation: KOST-Val reads a JP2 file (ISO 15444) and uses Jpylyzer to validate the structure and the content.

JPEG validation: KOST-Val reads a JPEG file (ISO 10918-1) and uses BadPeggy to validate the structure and the content⁵.

SIP validation: KOST-Val reads an SIP (eCH-0160⁶ v1 and v1.1 as well as Swiss Federal Archives SFA v1⁷ and v4⁸) and validates the mandatory requirements of the SIP specification. The validated requirements are organised into groups such as folder structure, schema validation, and checksum validation. At the outset, a file format validation is performed.

¹ The specification can be downloaded from the eCH website:

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

² The restriction is related mainly to the maximum throughput by 72,000 pages per year. More information on the license see Chapter 11.1. This restriction may be lifted if the 3-Heights™ PDF/A Validator API license by PDF-Tools is acquired and activated with the license manager.

³ ISO19005 does not stipulate that the images contained must be valid. The validation of the images is an archival requirement, especially in the use of PDF/A as an image format.

⁴ On the problem of the JBIG2 compression see http://kost-ceco.ch/cms/index.php?jbig2-compression_de. The KOST Preservation Planning Expert Group PPEG recommends to renounce the compression type JBIG2 when creating PDF files until further notice.

⁵ KOST-Val further evaluates and interprets the error message "Not a JPEG file".

⁶ The specification can be downloaded from the eCH website:

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

⁷ The specification can be downloaded from the SFA website:

<http://www.bar.admin.ch/themen/00876/00877/index.html?lang=en>.

⁸ The specification can be downloaded from the SFA website:

<http://www.bar.admin.ch/themen/00876/00877/01561/index.html?lang=en>.

The results (including information on inconsistencies and errors) are output for every step and written into a validation log.

The validation steps are executed sequentially. Whenever possible the validation shall continue after an error has been detected in order to reduce the number of correction cycles.

Please refer to the Annex for more detailed information on the different formats and validation steps.

2 System requirements

Microsoft Windows 98 or later

128 MB RAM or more (512 MB or more is recommended)

20 GB disk space or more

Java Runtime Environment (JRE) Version 6 (resp. 1.6)

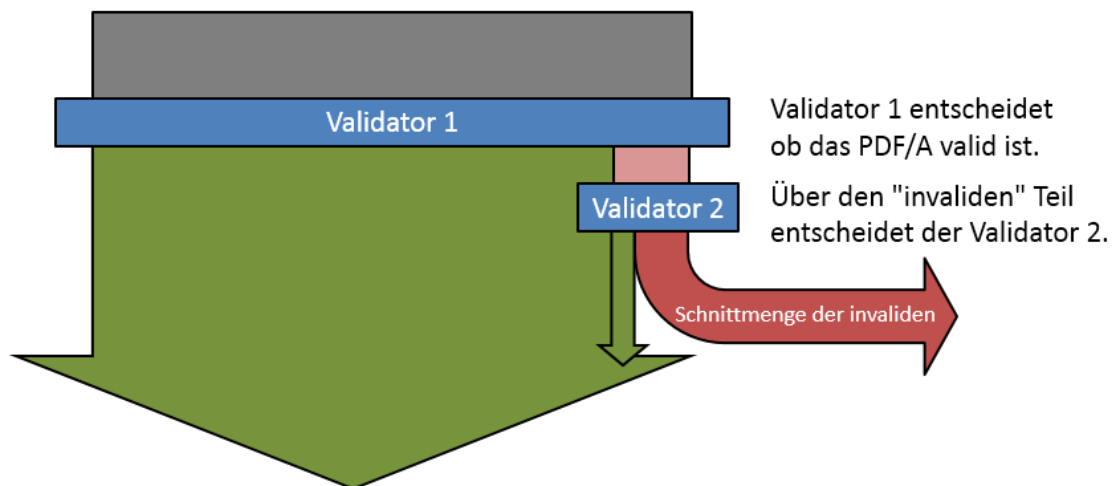
3 Open issues / Feedback

Open issues ranging including bugs, requested features, and questions, are listed on the software development platform GitHub at <https://github.com/KOST-CECO/KOST-Val/issues> and can also be communicated to kost-val@kost-ceco.ch.

These issues are managed by the development team. Any and all contributions are welcome.

4 Einleitung duale PDF/A-Validierung [translation in progress]

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.⁹



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

5 Diagnosedaten [translation in progress]

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

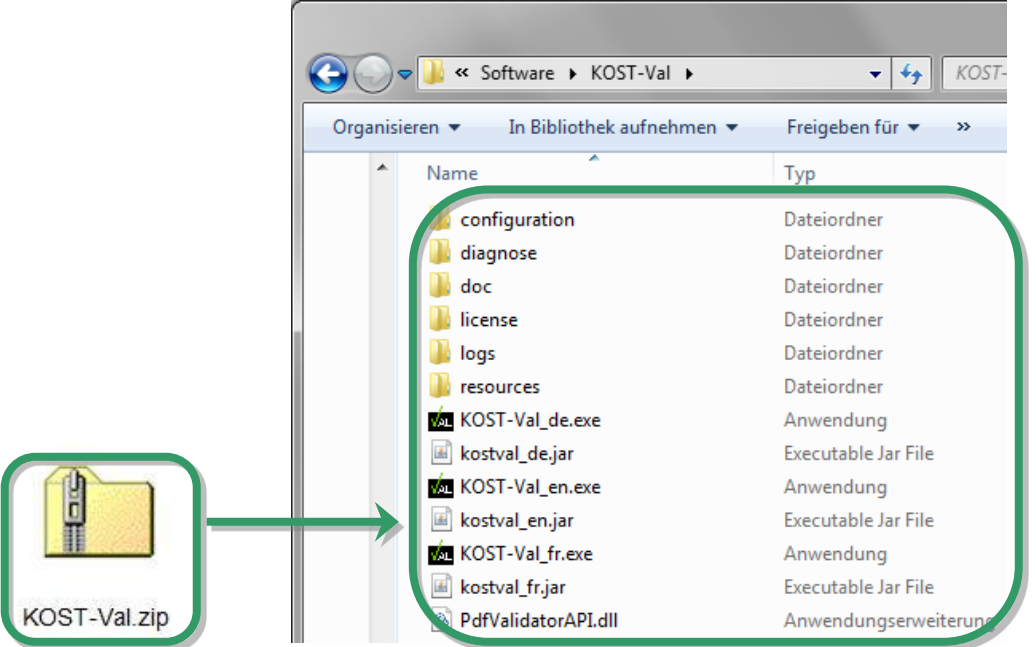
Wir sind Ihnen dankbar, wenn Sie diese Diagnosedatei bei Gelegenheit der KOST-Geschäftsstelle (kost-val@kost-ceco.ch) zusenden könnten.

⁹ 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.

6 Installation

6.1 Installation of KOST-Val

1 Download KOST-Val and unpack the ZIP file into the relevant folder.



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

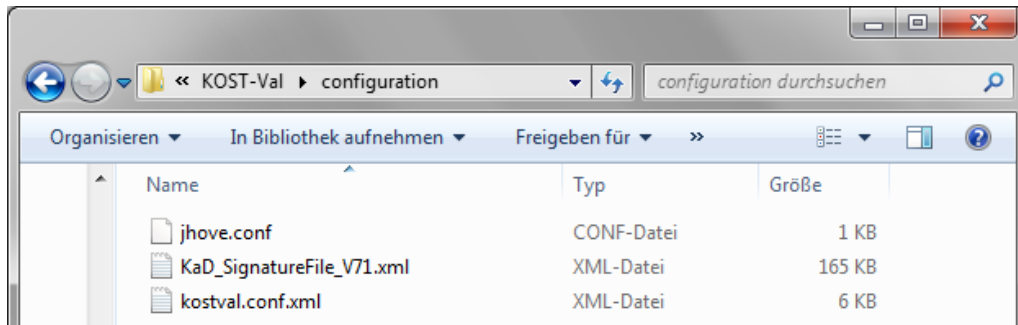
6.2 Optional installation of third-party applications

6.2.1 Installation of PDF/A Manager by PDFTron

! KOST-Val can also make indirect use (via console call) of the commercial software PDF/A Manager by PDFTron in order to validate PDF/A files. A trial version can be downloaded from <http://www.pdftron.com/downloads.html> for testing purposes. If PDF/A-Manager by PDFTron is not installed, the integrated 3-Heights™ PDF/A Validator by PDF-Tools is used.

7 Configuration of KOST-Val

All configurations of KOST-Val can be set in the configuration file "kostval.conf.xml" located in the „configuration" subfolder. Please use an XML editor or Notepad ++¹⁰ to modify XML files.



The „configuration" subfolder further contains the files "jhove.conf" and "KaD_SignatureFile_V71.xml" that are described in chapter 7.2.

7.1 Parts of the configuration file "kostval.conf.xml"

The configuration file "kostval.conf.xml" consists of several parts.

The pre-installed configuration allows immediate validation of PDF/A, TIFF, SIARD, JP2, JPEG and SIP.

The following is a short description of the configuration parts.

7.1.1 General

xml-Tag	Description: default value
<pathtologfile>	Path to log folder of KOST-Val: logs
<pathtoworkdir>	Path to temporary working directory of KOST-Val: temp_KOST-Val
<pathtodiagnose>	Path to the directory of the diagnostic data (see also chapter 5): diagnose
<pathtojhoveconfig>	Path to included JHOVE configuration file: configuration\jhove.conf

7.1.2 PDF/A

xml-Tag	Description: default value
<pdfavalidation>	Switch for PDF/A validation: yes
<firstvalidator>	Indicates which Validator PDFTools or PDFTron primarily should perform the PDF/A validation: PDFTools
<dualvalidation>	Indicates whether a dual or a simple validation should be performed. (see also chapter 0): simple
<pathtopdftronexe>	Path to PDFTron (see also chapter 6.2.1): C:\Software\pdfa\pdfa.exe

¹⁰ Portable Notepad ++ is available from http://portableapps.com/de/apps/development/notepadpp_portable.

xml-Tag	Description: default value
<pdfa1>	Specification of minimum conformity level for version 1: 1B
<pdfa2>	Specification of minimum conformity level for version 2: 2B
<pdfaimage>	Indicates if the image validation (JPEG and JP2) should be performed: yes
<jbig2allowed>	Indicates if the JBIG2 compression ¹¹ is allowed: no

7.1.3 SIARD

xml-Tag	Description: default value
<siardvalidation>	Switch for SIARD validation: yes

7.1.4 JP2

xml-Tag	Description: default value
<jp2validation>	Switch for JP2 validation: yes

7.1.5 JPEG

xml-Tag	Description: default value
<jpegvalidation>	Switch for JPEG validation: yes

7.1.6 TIFF¹²

xml-Tag	Description: default value
<tiffvalidation>	Switch for TIFF validation: yes
<allowedcompressionX>	Indicates whether compression algorithm X is allowed. By default the following values are allowed: uncompressed, CCITT 1D, CCITT Group 3, CCITT Group 4, LZW, PackBits
<allowedphotointerY>	Indicates whether colour space Y is allowed. By default the following values are allowed: white is zero, black is zero, RGB, palette color
<allowedbitspersampleZ>	Indicates whether Z bits per sample are allowed. By default the following values are allowed: 1, 4, 8, 16
<allowedmultipage>	Indicates whether multipage TIFF is allowed: 1
<allowedtiles>	Indicates whether tiled images are allowed: 0
<allowedsize>	Indicates whether file sizes exceeding 1'000'000'000 Bytes (~1GB) are allowed: 0

¹¹ On the problem of the JBIG2 compression see http://kost-ceco.ch/cms/index.php?jbig2-compression_de. The KOST Preservation Planning Expert Group PPEG recommends to renounce the compression type JBIG2 when creating PDF files until further notice.

¹² The default values are taken from KOST's Preservation Planning Recommendation (http://kost-ceco.ch/cms/index.php?preservation_de). This uses the Baseline TIFF specification to which it applies restrictions and extensions.

7.1.7 SIP

xml-Tag	Description: default value
<allowedlengthofpaths>	Maximum number of characters in file paths: 179
<allowedlengthoffiles>	Maximum number of characters in file names: 40
<allowsipname>	Specification of the structure of SIP names: SIP_[1-2][0-9]{3}[0-1][0-9][0-3][0-9]_\\w{3}
<pathtodroidsignature>	Path to appropriate DROID signature file: configuration\KaD_SignatureFile_V71.xml
<allowedversionbar1>	Indicates whether SFA SIP specification, version 1, is allowed: 1
<allowedversionbar4ech1>	Indicates whether eCH-0160, version 1 (= SFA SIP specification, version 4) or version 1.1, are allowed: 1
<allowedformats>	Lists the allowed file formats. The list consists of sub-elements with detailed information. The following are allowed per default: TXT, TIFF, JP2, PDFa1, PDFa2, WAVE, CSV, SIARD

7.2 "KaD_SignatureFile_V71.xml" and "jhove.conf"

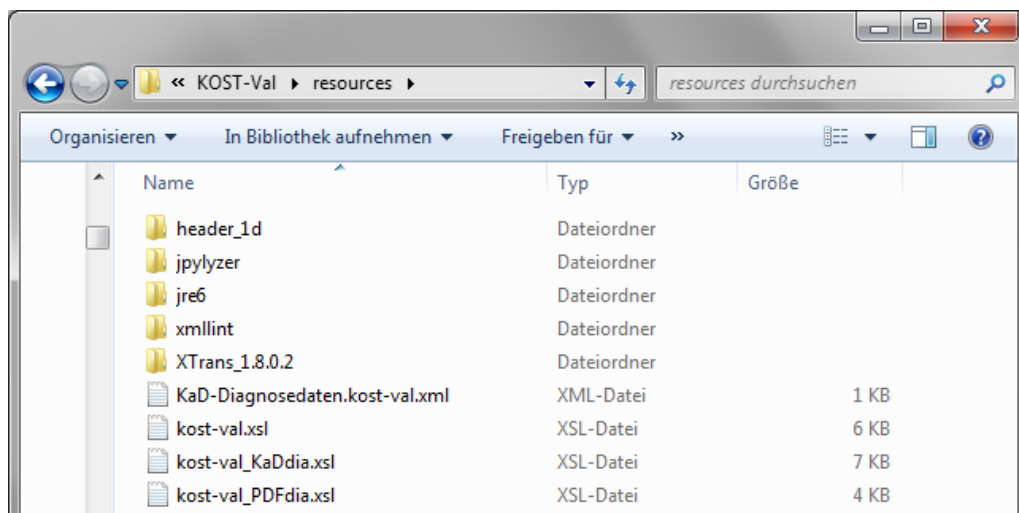
The "configuration" folder also contains the files "KaD_SignatureFile_V71.xml" and "jhove.conf" that do not need adjustment.

"KaD_SignatureFile_V71.xml" is used for format recognition. It is based on DROID, version 28.08.2013, and has been adapted by KOST¹³.

"jhove.conf" is used for the internal validation by JHOVE.

8 Resources of KOST-Val

All resources of KOST-Val are stored in the subfolder "resources".



¹³ KOST-Val uses KaD-SignatureFile by KOST instead of the DROID SignatureFile (see https://github.com/KOST-CECO/KaD_SignatureFile). Both files are compatible. KaD-SignatureFile comprises only the formats analysed in the KOST Catalogue of archival file formats KaD and permits their recognition in the granularity recommended by KOST. It leads to a significant increase in efficiency as compared to the DROID SignatureFile and thus enhances usability.

9 Start the validation

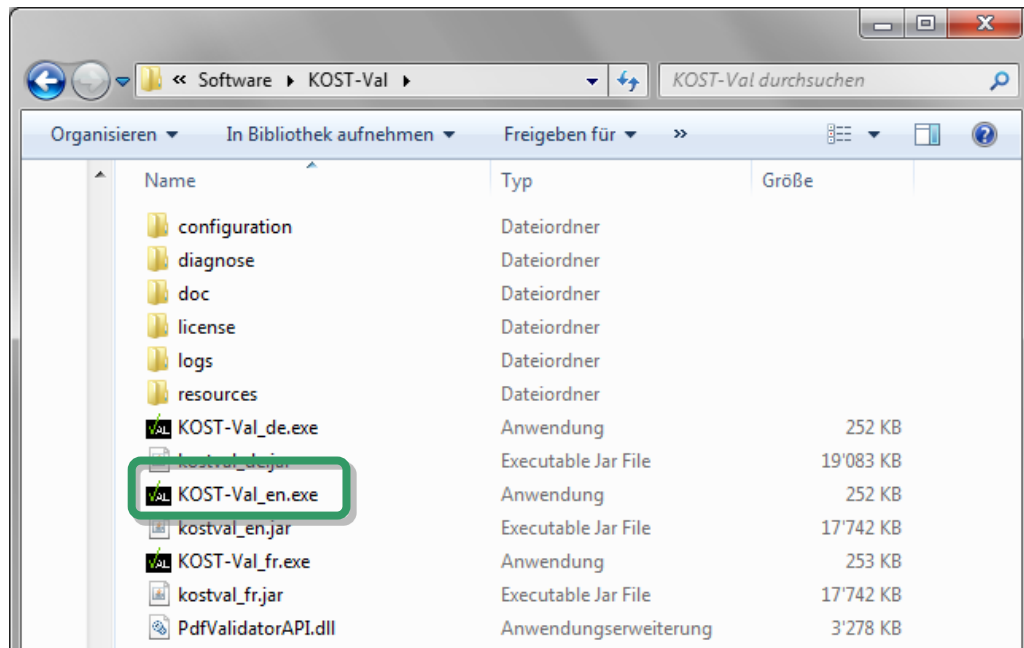


KOST-Val is not thread safe!

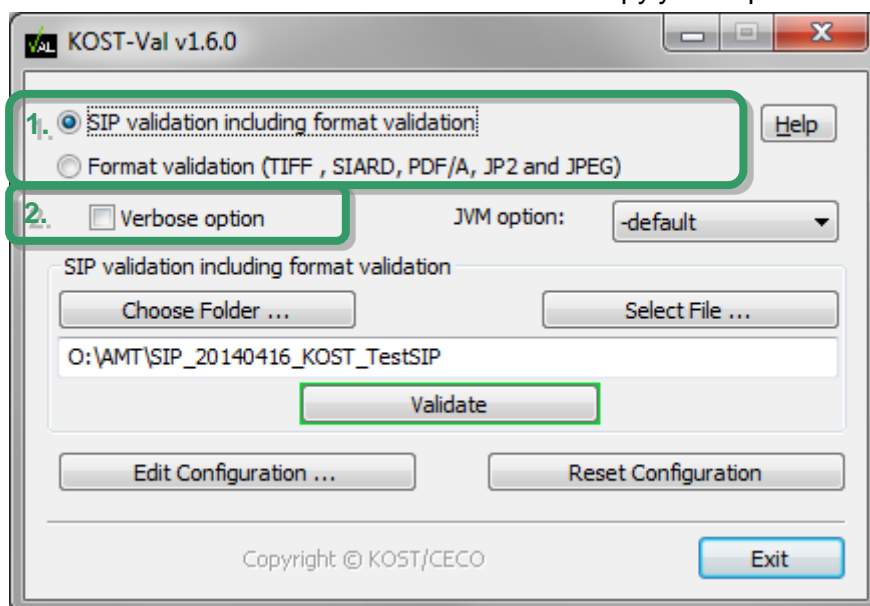
That is to say that concurrent instances of KOST-Val cannot be executed without interfering with each other. Concurrent execution of KOST-Val may lead to errors such as a missing working copy.

9.1 Start the validation using the "KOST-Val_en.exe" GUI

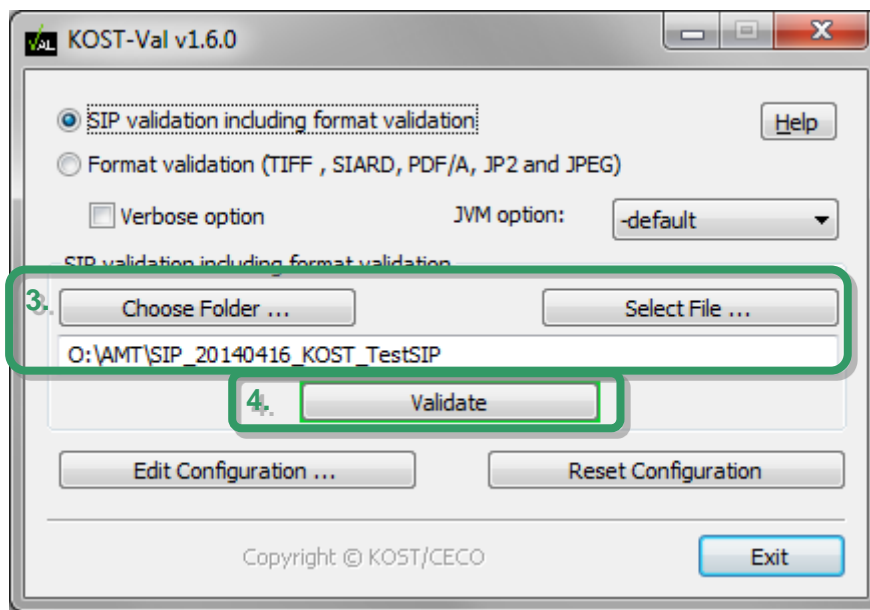
- 1 Double click on „KOST-Val_en.exe“ in the folder „KOST-Val“ to start KOST-Val.



- 2
 1. Choose whether to validate only formats or to validate a SIP including the formats.
 2. Choose whether to retain the JHOVE and Jpylyzer report.



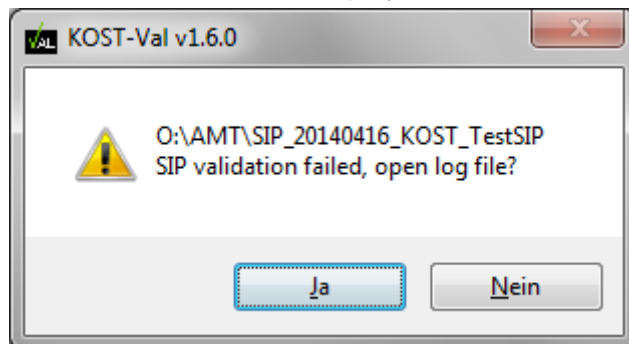
3. Choose or input the path to the file to be validated.
4. Start the validation.




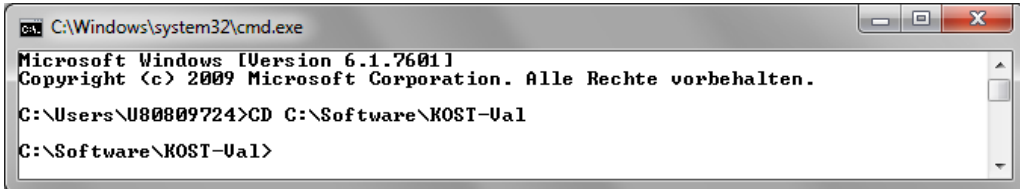

Hint:

If required, the default Java Virtual Memory can be quickly adapted. -Xmx should be adjusted in "Out of Memory" and -Xss at "Stack Overflow" errors. The configuration can be adapted temporarily. The temporary configuration is reset to the default values by clicking on the "Exit" or "Reset Configuration" buttons.

- 3 The validation result is displayed, and the KOST-Val log file can be opened.



9.2 Start the validation manually

1	<p>Prepare the folder or file to be validated.</p> 
2	<p>Open a command prompt (Start → Execute... → cmd [OK]) and change to the desired working directory (CD C:\Software\KOST-Val)¹⁴.</p> 
3	<p>Invoke the KOST-Val command (separate command options with spaces).</p>  <p> A resources\jre6\bin\java.exe¹⁵ -jar B kostval_en.jar C --format (or --sip for SIP validation) D O:\AMT\Dateisammlung or O:\AMT\Dateisammlung\jim__ah.tif E nothing or -v (verbose) </p> <p> A = Java command (resources\jre6\bin\java.exe -jar), B = relative path and file name kostval_en.jar, C = --format (or --sip for SIP validation), D = relative path and file name of the folder or file to be validated, E = verbose option to retain the original reports. </p> <p><u>Notes:</u> A command component that contains spaces needs to be enclosed in quotation marks. </p>

¹⁴ To change the drive type, e.g., CD /D.

¹⁵ Invoking `java -jar` is possible only if Java Runtime Environment (JRE) version 6 is the standard version. JRE7 can cause crashes. KOST-Val is massively slower in connection with JRE8.

KOST-Val can be invoked from any location. However this may require using absolute paths in both the program call and the configuration file "kost-val.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_en.jar --format O:\AMT\Dateisammlung

```

- A** C:\Software\KOST-Val\resources\jre6\bin\java.exe -jar
- B** C:\Software\KOST-Val\kostval_en.jar
- C** --format (or --sip for SIP validation)
- D** O:\AMT\Dateisammlung or O:\AMT\Dateisammlung\jim__ah.tif
- E** nothing or -v (verbose)

- 4** The file has been validated as soon as "Valid" or "Invalid" is displayed in the command window. The folder has been validated as soon as the prompt (C:\Software\KOST-Val>) is displayed.

```

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

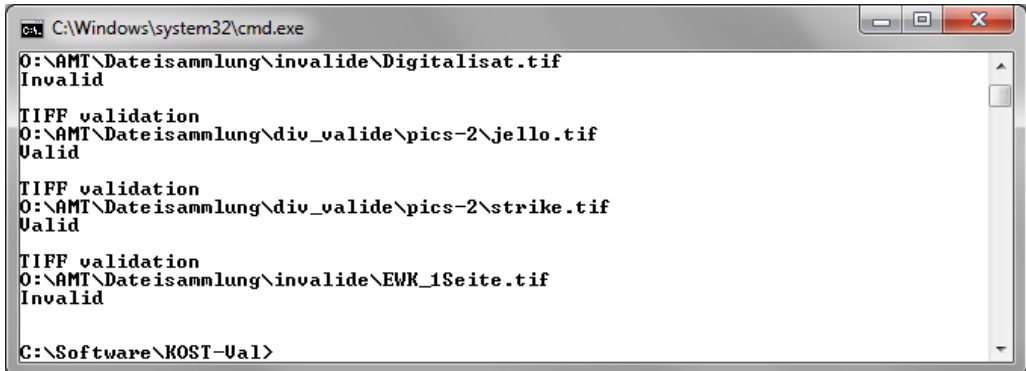
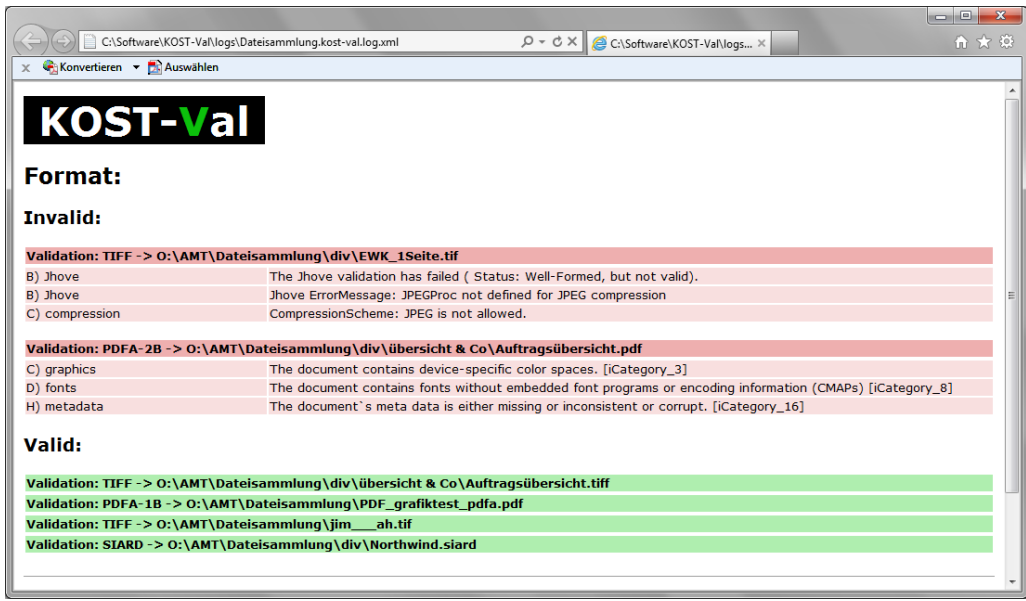
```

Detailed results are available in the file kost-val.log.xml.

The overall result (valid/invalid file) is output as well. In addition, it is visible in the program's exit status in order for the validation to be embedded into an automated process chain. The exit status can take the following values:

- 0 everything is ok
- 1 incorrect program call
- 2 not valid

10 Interpret the validation results

1	The validation steps are listed in chapter 12.1.
2	<p>KOST-Val first displays the validation type and the file name in the command prompt output. A validation that has completed without error returns “valid”. If a validation identifies an error it returns “Invalid”.</p>  <p>If the option „-v“ (verbose) has been activated, the original validation report is saved as well in the log folder. Since KOST-Val transmits all error messages without modification, this option is of interest only for the specialist user and in particular cases.</p>
3	<p>The log file lists additional detail on every invalid validation step in particular, the affected validation step and the corresponding error.</p> 

11 Copyright

KOST-Val has been developed by KOST. All rights reserved. KOST-Val has been published by KOST in 2012 under a GNU General Public License v3+.

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

KOST-Val uses the following unmodified components of other manufacturers by embedding them directly into the source code:

Third party application / component	Version	License
3-Heights™ PDF/A Validator API http://www.pdf-tools.com	4.5.6	see Chapter 11.1
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
iText http://itextpdf.com/	5.5.5	AGPL v3 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 uses the following unmodified components of other manufacturers which are delivered with KOST-Val:

Third party application / component	Version	License
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_plug-in	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

Users of KOST-Val are requested to adhere to these components' terms of licence available in the folder KOST-Val\license.

11.1 3-Heights™ PDF/A Validator API-Lizenz [translation in progress]

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, 2.10.2014

PDF Tools AG


Dr. Hans Bärffuss

Bern, 28. 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.

11.2 PDF/A Manager by PDFTron

KOST-Val can also make indirect use (via console call) of the commercial software PDF/A Manager by PDFTron in order to validate PDF/A files. This software is not part of KOST-Val. Users wishing to validate PDF/A files with KOST-Val are requested to purchase and install PDF/A Manager separately. Please refer to chapter 6.2.1 for further information.

12 Annex

12.1 Program structure

KOST-Val is structured according to the following requirements:

Functional requirements:

TIFF validation: KOST-Val reads a TIFF file and uses JHOVE to validate the following:

Validation step	Description
A (exit on error)	Recognition
B	Jhove
C	Compression
D	Colour space
E	BitsPerSample
F	Multipage
G	Tiles
H	File size

SIARD validation: KOST-Val reads a SIARD file and validates the following:

Validation step	Description
A (exit on error)	Readability
B (exit on error)	Primary folder structure
C (exit on error)	Validation of header
D (exit on error)	Validation of structure
E	Validation of table columns
F	Validation of table rows
G	Validation of tables
H	Validation of content
I	Recognition of SIARD
J	Additional primary data

PDF/A validation: KOST-Val reads a PDF file and uses 3-Heights™ PDF/A Validator by PDF-Tools or PDF/A Manager by PDFTron to validate the following:

Validation step	Description
A (exit on error)	General
B	Structure
C	Graphics
D	Fonts
E	Transparency
F	Annotations
G	Actions & Interactions
H	Metadata
I	Accessibility
J (configurable)	Image & JBIG2

JP2 validation: KOST-Val reads a JP2 file and uses Jpylyzer to validate the following:

Validation step	Description
A (may exit on error)	Recognition and Jpylyzer
B	Metadata
C	Image
D	Extended

JPEG validation: KOST-Val reads a JPWG file and uses BadPeggy to validate the following:

Validation step	Description
A (may exit on error)	Recognition and BadPeggy
B	Corrupt data
C	Invalid file structure
D	Other problems

SIP validation: KOST-Val reads an SIP file and validates the following requirements of the SIP specification:

Validation step	Description (name of step)
1a (exit on error)	Readability
1b (exit on error)	Primary folder structure
1c (exit on error)	Folder and file names
1d (exit on error)	Schema validation of metadata.xml
1e	Determine type of SIP
1f	Primary data in folder
2a	Missing primary data
2b	Additional primary data
2c	Validation of checksums
2d	Recording of GEVER primary data
3a	Format recognition
3b	Additional formats
3c	Format validation
3d	Validation of range of dates

For every step the results (including information on inconsistencies and errors) are output and written into a validation log.

The overall result (valid/invalid file) is output as well. In addition, it is visible in the program's exit status in order for the validation to be embedded into an automated process chain. The exit status can take the following values:

- 0 everything is ok
- 1 incorrect program call
- 2 not valid

The validation steps are executed sequentially. Whenever possible the validation shall continue after an error has been detected in order to reduce the number of correction cycles.

Non-functional requirements:

External programs or java frameworks are used for particular tasks.

The application has a modular structure that allows for inserting additional validation modules without further ado.

The validation log and exit status permit an easy readout of a single validation result and allow the utilisation of the tool in a process chain.

The console output is limited on the validation module, the final results of either “valid” or “invalid” and the path to the file. All additional information is documented in the log file.

12.2 Functional Principle of Format Validation

