

Ach, wie gut, dass niemand weiß, wie mein Dateiformat nun heißt

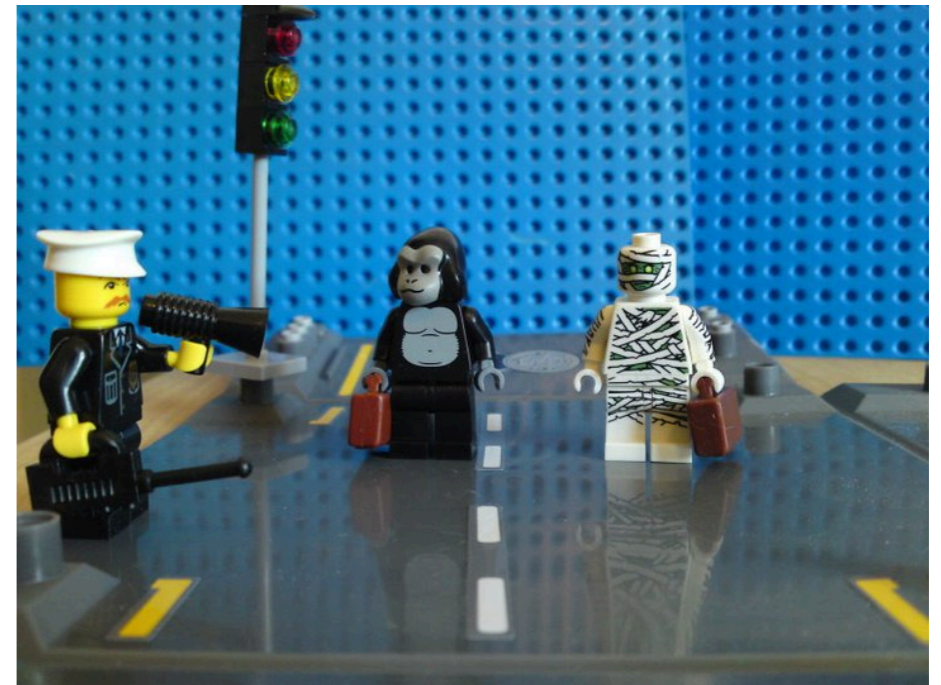
DROID und JHOVE

Yvonne Tunnat

Digitale Langzeitarchivierung

106. Deutscher Bibliothekartag

30.05.2017



Überblick

Theoretischer Teil

- Warum Formaterkennung?
- Vorstellung DROID (Digital Record Object Identification)
- Warum Formatvalidierung?
- Vorstellung JHOVE (JSTOR/Harvard Object Validation Environment)
- nestor – das deutschsprachige Kompetenznetzwerk zum Thema digitale Langzeitarchivierung
- Die nestor-AG Formaterkennung

Hands-On-Teil:

- Formaterkennung mit DROID
 - Formatvalidierung mit JHOVE: PDF, TIFF, JPEG
 - Bonus: Mögliche Zuarbeit zu PRONOM
-

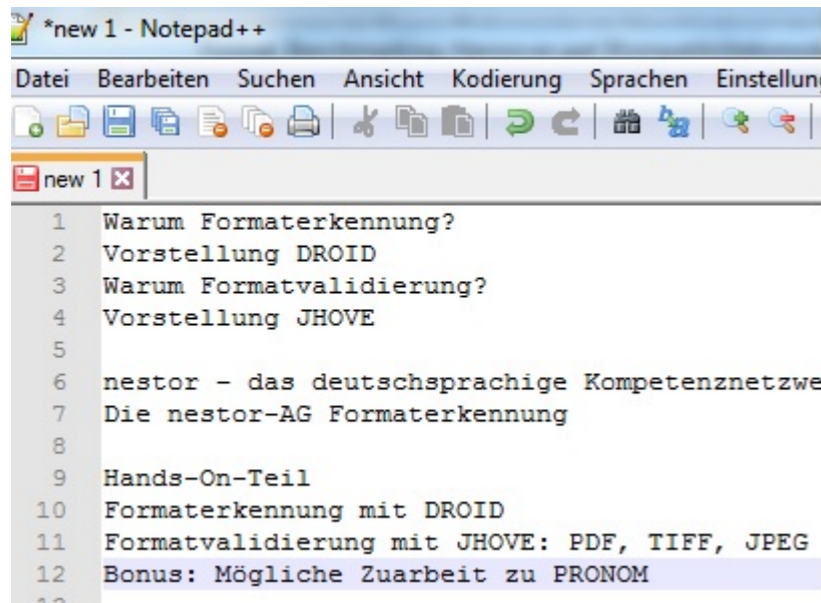
Warum Formaterkennung?

- Pro Jahr kommen ca. 90 Dateiformate dazu (Quelle)
- Das Formaterkennungstool mit der größten Formatbibliothek (TrID) unterscheidet zwischen 8500 verschiedenen Dateiformaten (TrID beinhaltet nur Binärdateiformate - dazu kommen noch etliche plain/text-Formate)
- Dateiformat entscheidet über Anzeige- und Bearbeitungstools (Womit kann ich die Datei öffnen?)
- Langzeitarchivierung: Dateiformat ist Grundlage für Risikomanagement und Preservation Planning, um die Langzeitverfügbarkeit zu sichern, beispielsweise durch Migration in aktuelle Dateiformate

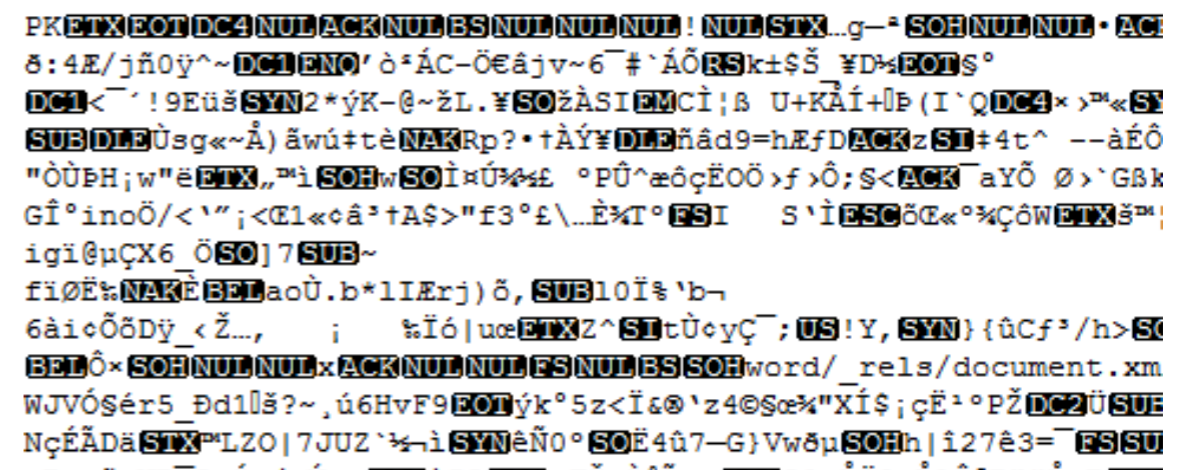
Was sind Binärdateiformate?

Keine reine Textdatei. Enthält nicht nur druckbare Zeichen. Enthält Bitmuster, ggf. Bilder etc.

Ansicht Plain-Text Datei im Texteditor



Ansicht Binärdatei (Word) im Texteditor



DROID

- Erstmals veröffentlicht 2006
- Nutzt die PRONOM Signature Files (aktuell 1447 verschiedene Signaturen)
- Entwickelt von der TNA (The National Archives) mit Zuarbeit der gesamten LZA-Community
- Ist das meistgenutzte Tool in der LZA-Community

Drei verschiedene Nutzungsmöglichkeiten:

- GUI (Graphical User Interface)
- via Kommandozeile (somit als Batch-Programm)
- Java library

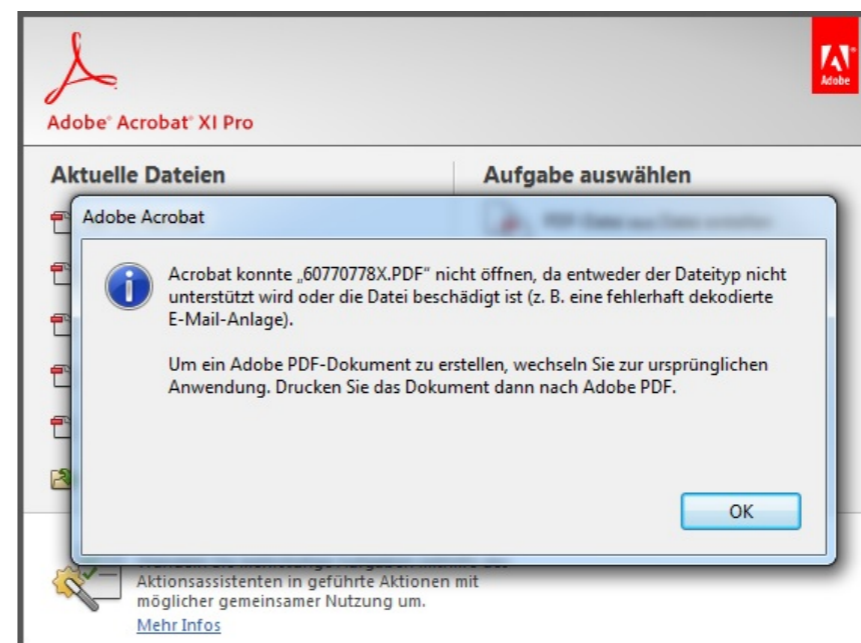
[User Guide](#)

Warum Formatvalidierung

Ich kenne nun das Dateiformat. Aber ist es das wirklich?

Validierung: Prüfung, ob Datei wirklich den Formatspezifikationen entspricht.

Ist eine Datei invalide, kann dies Auswirkungen auf die Darstellbarkeit haben (=„Datei kaputt“).



JHOVE

- Erstmals veröffentlicht 2005 von der Harvard University Library
- Seit März 2014 gehostet bei der OPF (Open Preservation Foundation)
- Community hilft mit via GitHub, Document Interest Group, JHOVE Hack days
- JHOVE kann 12 verschiedene Formate validieren, u. a. PDF, TIFF, JPEG

Drei verschiedene Nutzungsmöglichkeiten:

- GUI (Graphical User Interface)
- via Kommandozeile (somit als Batch-Programm)
- Java library

nestor

- Seit 2003 Deutsches Kompetenznetzwerk zur digitalen Langzeitarchivierung
- 20 Partner mit 11 aktiven AGs

nestor AG Formaterkennung

- Aktiv seit 2014
- 13 Institutionen, 6 Unterarbeitsgruppen
- Öffentlich sichtbare Wissensdatenbank (im Aufbau)

Weiterführende Links

[Wiki der nestor-AG Formaterkennung](#)

[DROID von The National Archives](#)

[JHOVE \(gehostet von der Open Preservation Foundation\)](#)

Weitere Formaterkennungstools:

[Siegfried](#)

[Fido](#)

[File](#) (Fine Free File Command)

[TrID](#)

[Apache Tika](#)

[Gvfs-info](#)

Hands On



Inhalt Stick

- Informationen zum Workshop und zur Dozentin (Abstract + Kurzvita)
- Dieser Vortrag (Bibtag2017_Formaterkennung.ppt)
- AnwendungDROID
- AnwendungJHOVE
- Dateisamples (beinhaltet verschiedene Ordner)

Anwendung DROID - GUI

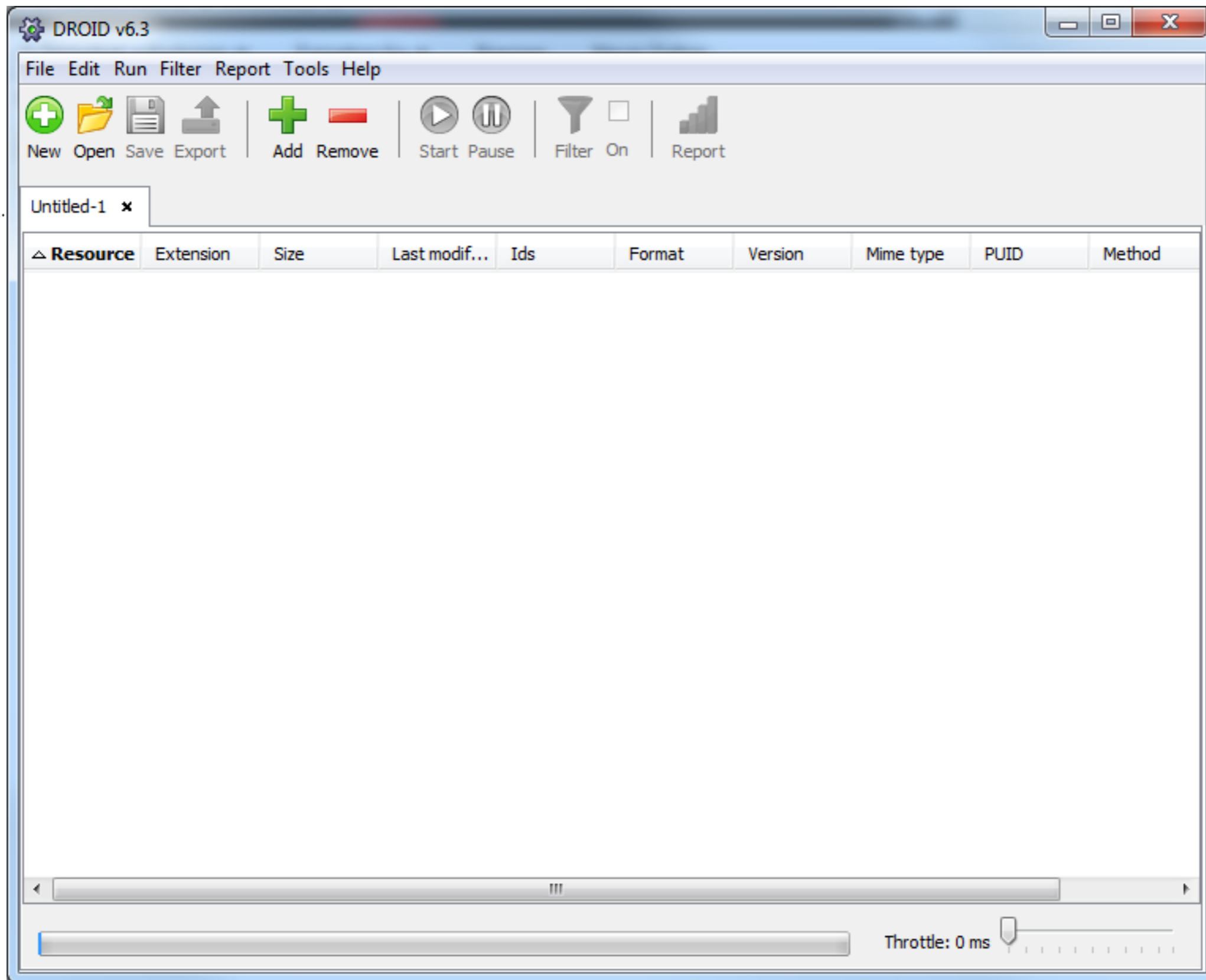
USB Stick:

AnwendungDROID_6_3_Sig90

Droid-binary-6.3-bin

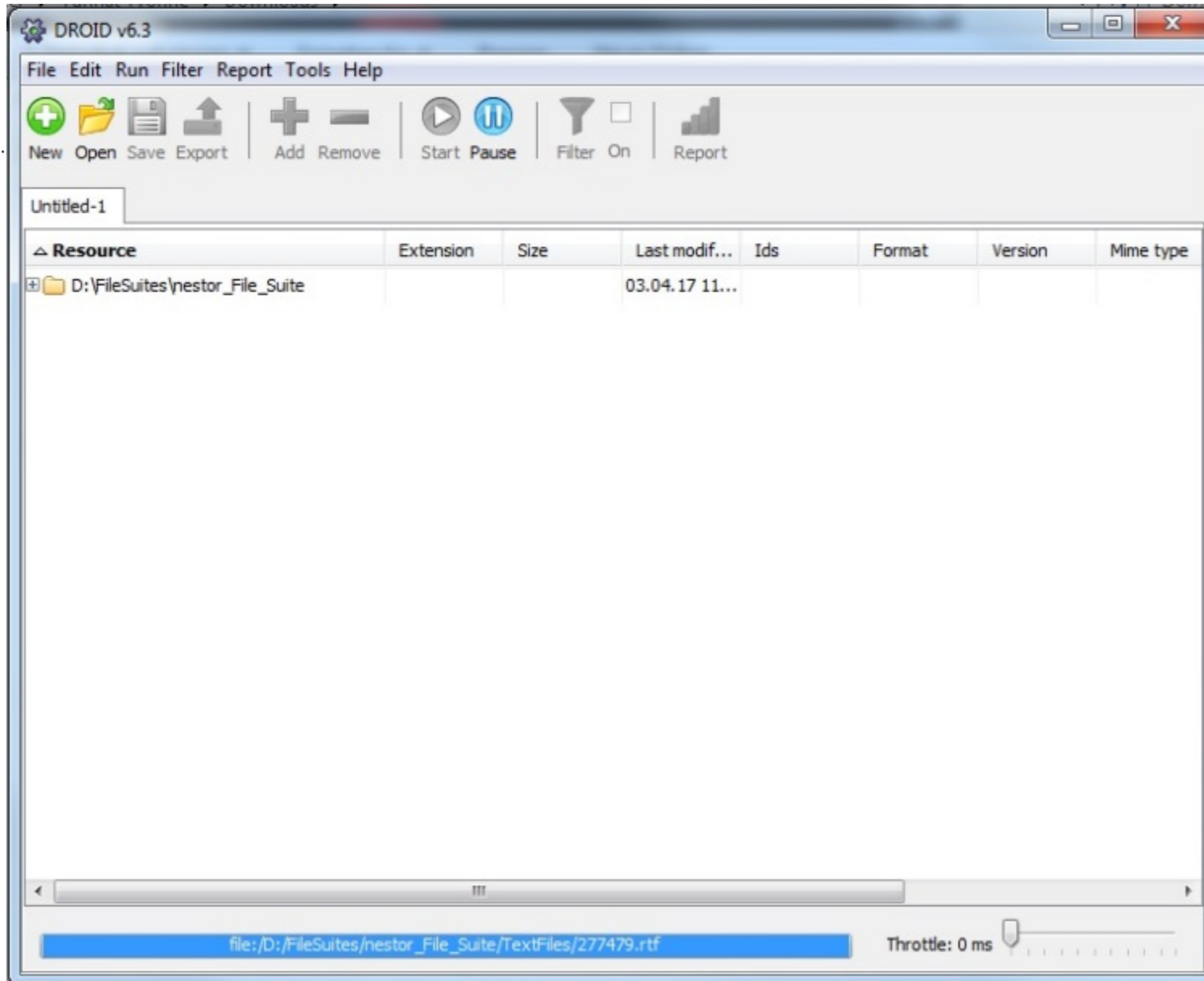
Droid-ui-6.3.jar (*Klick zum Starten von DROID*)

(Alternativ download auf der TNA Seite)



Bedienelemente

- Via „add“ zu dem Ordner oder der Datei navigieren, die analysiert werden soll.
- Open hat mit DROID Profilen zu tun
 - DROID-Profil: abgespeicherte Konfiguration (sinnvoll z. B. bei der CMD-Nutzung)
- New öffnet neuen Tab innerhalb von DROID
- Start beginnt die Untersuchung



Output

Ich nutze eigentlich immer

- Alles markieren (Strg + A)
- Copy to Clipboard (Strg + C)
- In Excel einfügen (Strg + V)

Die Report-Funktionen nutze ich nicht

Output in Tabellenform

- **Resource:** Pfad zur Datei / Dateiname
- **Extension:** Zeichenfolge, die hinter dem letzten „.“ im Dateinamen steht sofern „.“ vorhanden
- **Size:** Dateigröße
- **Last modified**
- **Format**
- **Version**
- **Mime Type**
- **PUID** (PRONOM's Persistent Unique Identifier)
- **Method** (Signature, Extension oder Container)

Exkurs: Was ist ein Mime-Type?

„Multipurpose Internet Mail Extensions“

Allgemeiner Datentyp, z. B.:

`text/plain`

`text/html`

`application/xml`

`image/gif`

Unterschiede in der Granularität der Erkennung, Bsp:

`application/pdf`

Für jedwede PDF-Version und „Geschmackssorte“ (PDF/A usw.)

Exkurs: Was ist eine PUID?

PRONOM's Persistent Unique Identifier

genauer Dateityp mit Version und „Geschmackssorte“ (z. B. PDF 1.7).

Auflösung der PUIDs auf dem Stick unter „hilfreiches -> PRONOM Signatures v90“

- Aufbau: Präfix „fmt“ oder „x-fmt“ + Zahl
- Jede PUID bleibt persistent, als deprecated (veraltet) bezeichnete PUIDs werden nicht neu vergeben, sondern sind weiterhin dokumentiert und verweisen auf ggf. neue PUIDs für das jeweilige Format
- Es gibt aktuell (PRONOM Signature File 90) 1493 PUIDs
- Das Format PDF hat mittlerweile 29 verschiedene PUIDs

Warum 29 verschiedene PDF- “Geschmacksrichtungen“?

Warum ist es für die LZA wichtig zu wissen, welches PDF genau ich vorliegen habe?

PDF 1.5 (fmt/19)

PDF/A-1b (fmt/354): Nicht „tagged“, erlaubt keine Ebenen und Transparenzen

PDF/A-2a (fmt/476): tagged, erlaubt Ebenen und Transparenzen

Übung

Testsuite Formaterkennung

Testsuite Formaterkennung ohne Extensions

Ggf. selbst mitgebrachte Samples analysieren

Ggf. andere Ordner auch untersuchen

Ergebnisse diskutieren

Diskussion: Wie gehe ich mit nicht erkannten Dateien um?

Was bedeutet „Dateiformat unbekannt“ für die Langzeitverfügbarkeit?

Beispiele aus der Übung / dem eigenen Berufsalltag

Long tail der Dateiformate – wenige Dateien erzeugen viel Arbeit

JHOVE (Version 1.16.3)

Anwendung JHOVE_1_16_3 -> jhove -> jhove-gui.bat

Ggf. über Edit -> select Module das entsprechende Modul auswählen, z. B. PDF-hul. Andernfalls besteht auch die Gefahr, dass ein invalides PDF einfach als „well-formed and valid Bytestream“ ausgezeichnet wird.

Via „File“ -> open oder einfach mit Drag und Drop eine Datei oder einen Ordner zum Analysieren auswählen

Output: auch speicherbar und mit Editor zu öffnen

Übung

PDF

TIFF

JPEG

GIF

In den jeweiligen JHOVE Modulen testen

Für die bessere Lesbarkeit des Output kann jhoveview.jar genutzt werden (siehe Stick)

Diskussion: Was bedeutet invalide?

Mögliche Auswirkungen auf die digitale Langzeitarchivierung

Beispiele aus den Übungen

Optional: Zuarbeit zu PRONOM

Kann sich das jemand hier vorstellen?

Falls das nicht besprochen werden soll, kann das hier nachgelesen werden:

Eigene Formatsignaturen für DROID und Zuarbeit zu PRONOM
<https://wiki.dnb.de/pages/viewpage.action?pageId=115213928>

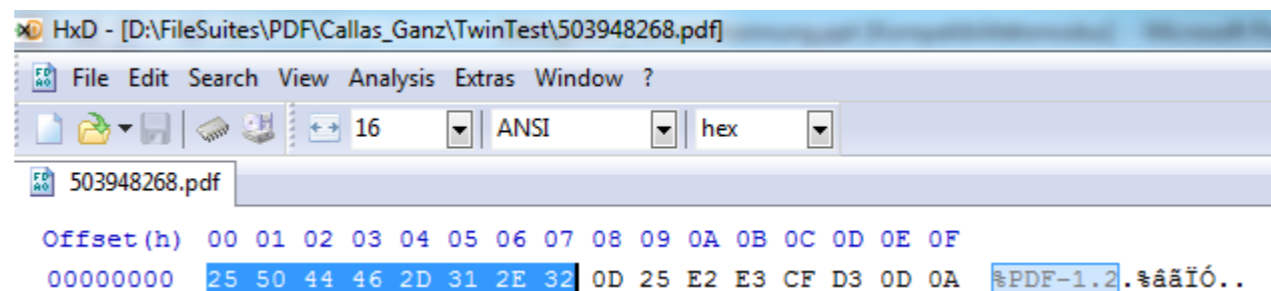
DROID (alle PRONOM-basierten Tools)

Formaterkennung durch Bitmuster

Hex-Editor (USB Stick unter Hex_Editor -> HxDen -> HxD.exe)

Formatidentifizierung mittels „magic Number“, „Signatur“
„Charakteristisches Bitmuster“.

Beispiel PDF:



The screenshot shows the HxD hex editor interface. The title bar indicates the file path: [D:\FileSuites\PDF\Callas_Ganz\TwinTest\503948268.pdf]. The menu bar includes File, Edit, Search, View, Analysis, Extras, and Window. The toolbar shows a search icon, a dropdown menu set to 16, and a dropdown menu set to hex. The file name 503948268.pdf is displayed in the address bar. The main window shows a hex dump with the following content:

```
Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
00000000 25 50 44 46 2D 31 2E 32 0D 25 E2 E3 CF D3 0D 0A PDF-1.2.      ..
```

Charakteristisches Bitmuster in einer Datei finden

Ein charakteristisches Bitmuster sollte zwei wesentliche Eigenschaften erfüllen:

- Das Muster muss in *allen* Dateien des zu beschreibenden Dateiformats vorkommen, bestenfalls immer an derselben Stelle (z.B. am Anfang der Datei).
- Das Muster sollte möglichst *nur* in Dateien des zu beschreibenden Dateiformats vorkommen, damit nicht ein anderes Dateiformat falsch identifiziert wird. Diese Eigenschaft kann natürlich nie restlos sichergestellt werden, denn das gewählte Muster könnte ja durchaus in einem anderen Dateiformat in einer anderen Funktion auftreten.

Wie findet man so ein Bitmuster?

Möglichkeit 1: In der Spezifikation (z. B. wäre dies beim PDF der Fall)


Möglichkeit 2: Anhand eines möglichst großen Samples des Dateiformats möglichst aus unterschiedlichen Quellen

Sofern das Bitmuster nicht gleich ins Auge fällt, gibt es hierfür Tools:
TrIDScan - Patterns scanner

Testen ob Signatur in DROID funktioniert

PRONOM: Signature Development Utility ([Link](#))

- Via Drop-Down-Menü „Anchor“ kann angegeben werden, wo in der Datei das Bitmuster zu finden ist.
- Die Signatur selber muss eingetragen werden, dazu der Name des Dateiformats.
- (PUID ist nur Platzhalter, wird vom TNA vergeben)
- Das Signature File kann dann intern via DROID getestet werden


 The National Archives

Prototype

PRONOM: Signature Development Utility

Name:	SQLite Database File Format		
Version:	3	Extension:	sqlite
PUID:	dev/1	Mimetype:	
Signature:	53514C69746520666F726D6174203300		
Anchor:	Absolute from BOF		
Offset:	0		
Max Offset:	0		

[Add Sequence](#) [Remove Sequence](#) [Save Signature File](#)

 Export to RDF [? Help](#)

PRONOM neue Signatur melden

Sofern die Tests zufriedenstellen waren, kann die neu erstellte Signatur gemeldet werden

<https://www.nationalarchives.gov.uk/contact-us/submit-information-for-pronom/>

Weiterbildung und Beteiligungsmöglichkeiten

- nestor Praktikertag am 28. Juni in Kiel:
- „Formaterkennung, Formatvalidierung und Tools“
- In Arbeit: JHOVE for Beginners Guide (Feedbackmöglichkeit)
- OPF Document Interest Group: Analyse der PDF-JHOVE-Fehlermeldungen hinsichtlich Impact, Cure und Testcase
- nestor AG Formaterkennung (bei Mitarbeit am Thema)