

ExiftoolGUI - „Arguments“-Datei verwenden

Haben Sie auch schon erlebt, dass Ihre DSLR-Kamera das aufgesetzte Objektiv nicht wirklich erkennt und in den Metadaten (Exifdaten) eine falsche Objektivbeschreibung liefert? Dann sind auch Sie ein Opfer der ungeordneten Vergabe von Identifikationsnummern für Objektiv-Typen geworden.

Seit der Einführung der Autofokus-Technologie werden Wechselobjektive elektrisch mit der Kamera verbunden und seither werden Objektive auch mit einem ROM-Chip ausgerüstet, der die Fokus/Blenden/Belichtungssteuerung unterstützt und zusätzlich eine Identifikationsnummer (ID) des Objektivtyps enthalten kann. Es wäre so schön gewesen, wenn sich die Branche sofort auf eine Norm geeinigt hätte, um die IDs nach Hersteller + Objektivtyp + Baureihe/Version unterscheiden zu können. Stattdessen kochte jeder Kamerahersteller sein eigenes Süppchen und nummerierte die Objektive nach Belieben durch. Manche vernachlässigten dies so weit, dass unterschiedliche Objektivtypen mit identischer ID vergeben wurden.

Ganz schwierig ist es bis heute für die Dritthersteller von Objektiven. Ohne kostspielige Lizenzierung beim Kamerahersteller werden die Objektiv-IDs nicht in deren Nummernfolge einbezogen. Die Dritthersteller behelfen und behelfen sich dadurch, dass sie häufig die LensID eines bauähnlichen Objektivs übernehmen, oder nur Füllnummern wie „128“ oder „255“ in den Objektivchip programmieren.

Das rächt sich inzwischen, denn mit fortschreitender Software-Entwicklung könnten immer mehr Aufgaben automatisiert werden. Adobe liefert das beste Beispiel in den 2011er-Updates von Lightroom und Photoshop. Anhand des Meta-Eintrags <LensType> kann die ID-Nummer zum Aufruf einer automatisierten Objektivkorrektur verwendet werden.

Noch gibt es hauptsächlich Lens-Profiles (Objektivprofile) für hochwertige Linsen, aber mit dem kostenlosen *Adobe-LensProfile-Creator* kann jeder Fotograf seine eigenen Profile erstellen—und für die Adobe-Community ins Netz stellen.

Es bedarf nur wenig Phantasie um vorher zu sagen, dass in den nächsten Jahren für jeden Glasscherben ein mehr oder weniger gutes Profil verfügbar sein wird, um geometrische Objektivverzerrungen, Chromatische Aberrationen und die Vignettierung automatisiert zu korrigieren.

Was bisher nur ein Schönheitsfehler war, nämlich die falsche Objektiverkennung wird dann Nachteil im Workflow der Bildbearbeitung von der Kamera, über den Raw-Converter zum fertigen Bild.

Metadaten in Kamera-Dateien

Es ist landläufig bekannt, dass eine Digitalkamera nicht nur das Bild sondern eine große Liste von Aufnahmedaten in die Aufnahme-datei schreibt. Diese werden unter dem Sammelbegriff Metadaten geführt und unterteilen sich in verschiedene Datenklassen. Die wichtigsten Klassen sind MAKER, EXIF und XMP.

In den MAKERNOTES übermittelt der Kamerahersteller Daten zur Kamera und zum erkannten Objektiv.

In den EXIF-Daten werden Informationen über die gespeicherte Aufnahme hinterlegt. Dazu gehört das Aufnahmedatum, Belichtungszeit, Blende, ISO, Aufnahmeprogramm und vieles mehr.

Im Zuge der weiteren Bildentwicklung und Bildbearbeitung werden vorhandene und Entwicklungsdaten (zusätzlich) in die XMP-Datengruppe geschrieben.

Fast jedes Grafikprogramm kann die wichtigsten Daten auslesen und bietet eine mehr oder weniger umfangreiche Anzeige der Metadaten an.

Spezielle EXIF-Viewer leisten mehr - und können je nach Arbeitslust des Programmierers fast alle Tags (Datenfelder) auslesen.

Die MAKERNOTES sind jedoch teilweise verschlüsselt und bleiben ein Geheimnis der Kamerahersteller. Deshalb heißt es dann bei manchen Tags „Inhalt unbekannt“.

Einige Programme bieten an, die Metadaten oder einen Teil der Daten zu editieren. Auch das geht mehr oder weniger flüssig und elegant vonstatten.

Exiftool

Das Programmpaket ist eine Entwicklung von Phil Harvey und wird unter einem Downloadlink vom Server der Queen's University in Canada kostenfrei angeboten.

Allerdings ist das „Kommandozeilenprogramm“ etwas für Puristen und Hobbyprogrammierer.

Dem Leid der vielen PC-Benutzer, die nur noch die Grafische Benutzeroberfläche (GUI=Graphical User Interface) von Windows oder Apple kennen, hat sich u.a. Bogdan Hrastnik angenommen, der die **exiftool-GUI** entwickelte, die sich über **exiftool** stülpt und einige Standardanwendungen in typischer Windows-Grafik-Umgebung zulässt.

Bogdan Hrastnik hat freundlicher Weise auch ein Kommandozeilenfenster eingebaut, in dem zusätzliche Aufgaben manuell verarbeitet werden können. Dabei erleichtert dieses Fenster die Aufgabe, denn es wird nicht die komplette Kommandozeile eingetragen, sondern nur noch der Befehlssteil mit den Anweisungen und Optionen. Das Kommando wird dann auf die Datei/en ausgeführt, die in einem Dateifenster markiert sind.

Eine weitere vorteilhafte Option ist die Möglichkeit, die zu ändernden Metadaten in einer „ARGUMENTE“-Datei zu hinterlegen. Diese wird dann in der Kommandozeile aufgerufen und abgearbeitet.

Doch genug der Vorrede, jetzt geht es an die Umsetzung. Ich setze voraus, dass Sie exiftool und exiftool-GUI herunter geladen und installiert haben (wenn nicht, dann lesen Sie zuerst das „Download & Install“-Kapitel am Schluss dieses Artikels).

Das Programm „exiftool.exe“ wird zwar im Programmpaket benötigt, aber in dieser Anwendung nicht direkt geöffnet.

Unsere Anwendung ist immer die *exiftoolGUI.exe*. Die ScreenCopy (unten) zeigt das Arbeitsfenster mit der Explorerfunktion zur Auswahl von Laufwerk, Verzeichnissen und Dateienliste.

In der Dateiliste können einzelne, mehrere oder alle Dateien markiert werden. Bei einer markierten Einzeldatei wird links unten die Bildvorschau und rechts der Auszug aus den Metadaten angezeigt.

Die Metadaten werden nach Gruppen sortiert. Die einzelnen Tags können per Mausklick rechts für die CUSTOM-Liste ausgewählt werden.

Unterhalb der Reiter für die Meta-Gruppen kann per Aufwärts-Pfeil ein angepasstes Editorenfenster geöffnet werden. Dahinter verbergen sich Edit-Funktionen, mit denen gängige Inhalte überarbeitet werden können. Eine

typische Funktion wäre hier die Berichtigung des Aufnahmedatums, wenn die Kamera nicht korrekt konfiguriert war.

Unten findet sich der Button <exiftool direct>. Mit diesem öffnet sich die Kommandozeile.

Im sichtbaren Beispiel wird die Zeile für die Bearbeitung der Objektivinformationen für ein Objektiv „QUANTARAY 70-300 mm“ angezeigt. Das ist ein Lizenzprodukt von Sigma für USA.

Die für das Objektiv zu bearbeitenden Optionen sind in der Datei „*QUANTARAY_70_300.args*“ hinterlegt.

Der Aufruf „Schreibe ObjDaten Quantaray...“ wird in der „exiftoolGUI.ini“-Datei als vordefinierte Aufgabe gespeichert.

Ist das einmalig geschehen, dann kann diese Aufgabe schnell und unkompliziert abgerufen werden.

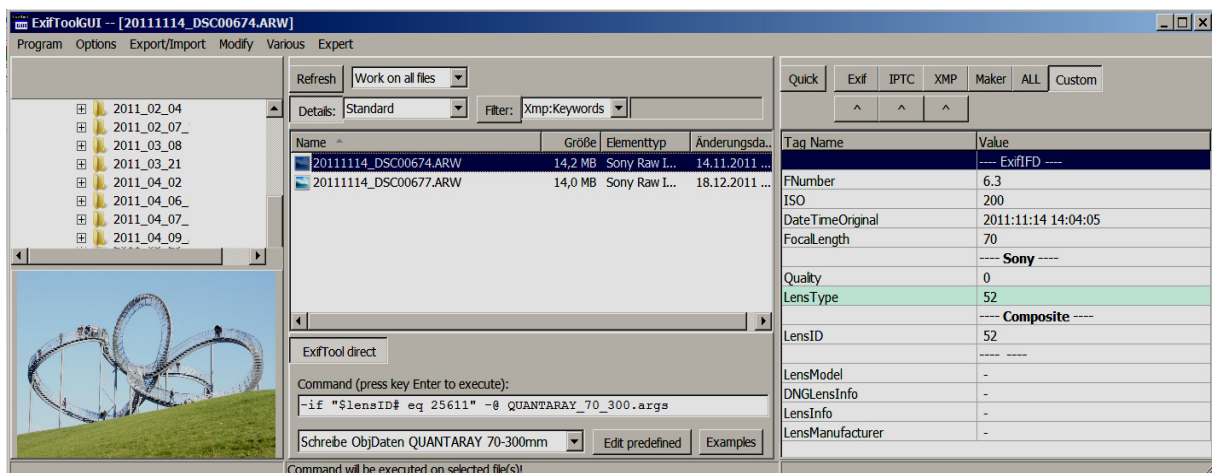
Die Kommandozeile enthält eine „-if“-Bedingung. D.h., die Aufgabe wird nur ausgeführt, wenn (engl.: „if“) die definierte Bedingung erfüllt ist. In diesem Fall muss die LensID „25611“ sein, um in der Datei entsprechende Änderungen vorzunehmen. Es können also bedenkenlos alle Dateien markiert werden. Sie werden nach der LensID gescannt, aber nicht verändert, wenn sie die Bedingung nicht erfüllen.

In der Beschreibung auf den folgenden Seiten werde ich nicht auf die Anwendung und Möglichkeiten von exiftoolGUI im Detail eingehen. Das kann jeder für sich an einigen Test-Duplikaten von Raw- und Bilddateien erarbeiten.

Hier soll nur die Anwendung der ARGUMENT-Dateien und die Einbindung in die INI-Datei beschrieben werden.

Dazu mehr auf den folgenden Seiten.

geo, dez 2011



Inhalt der Datei „QUANTARAY_70_300.args“

Die Arguments-Datei (`_args`) ist eine reine Textdatei, die mit einem beliebigen Text-Editor angelegt werden kann. Office-Programme wie WORD eignen sich nicht als Text-Editoren. Windows hat einen Texteditor im <Zubehör>. Komfortabler ist das Freeware-Tool „notepad++“.

Beim Schreiben einer eigenen „args“-Datei muss die Synopsis und Syntax der Programmiersprache - einschließlich von (nicht) vorhandenen Leerzeichen und führenden Bindestrichen - beachtet werden. # (Raute) zu Beginn einer Zeile oder nach dem Abschluss eines Kommandos bedeutet, dass es sich um eine Kommentarzeile handelt, diese Zeile (ab Position der Raute beschreibend ist, aber nicht ausgeführt wird. Im folgenden Beispiel wird z.B. <Sony:LensType> durch # deaktiviert, weil aktiv in der nächsten Zeile <LensType> auf „0“ gesetzt wird.

Wird die Raute innerhalb der Tag-Optionen verwendet, dann hat sie die Bedeutung, dass der Inhalt als „Zahl“ dargestellt werden soll. Das trifft auf die LensID zu, die mit „25611“ bzw. „0“ Ganzzahlenwerte enthält.

```
# Quantaray 70-300mm F4-5.6 hat die LensID 25611
# Die LensID wird auch von anderen Objektiven verwendet und an SONY-Kameras falsch als
# "Minolta 75-300 mm F4.5-5.6" beschrieben
# Adobe korrigiert anhand der LensType/LensID aus einer internen Datenbank zu „Minolta“,
# auch wenn die anderen beschreibenden Tags geändert sind.
# Das Problem ist nur lösbar, indem LensType/LensID auf "NULL" gesetzt werden und Adobe nur
# anhand der anderen Objektivdaten das Objektiv erkennt
# Speziell sind das die Daten für "kürzeste / längste Brennweite & Offenblende in den beiden
# Brennweite-Grenzen
# 18-12-2011 geo

-ExifIFD:UserComment=Quantaray 70-300 F4-5.6 LensID=25611
-ExifIFD:LensInfo=70/1 300/1 40/10 56/10
-ExifIFD:LensModel=70-300mm F4-5.6 (QUANTARAY)
# -Sony:LensType#=25611 <<< durch Raute deaktiviert
-Sony:LensType#=0
-xmp:LensManufacturer=SIGMA for QUANTARAY LensID 25611
-xmp:Lens=70-300mm F4-5.6 (QUANTARAY)
# -xmp:LensID#=25611 <<< durch Raute deaktiviert
-xmp:LensID#=0
-xmp:LensInfo=70/1 300/1 40/10 56/10
-DNGLensInfo=70/1 300/1 40/10 56/10
```

Zeilen wie <LensInfo> enthalten strukturierte Angaben über die kürzeste und längste Brennweite und die Offenblende bei kürzester und längster Brennweite. Im konkreten Fall beträgt die Offenblende bei 70 mm=F4 und bei 300 mm= F5.6. Da in dieser Zeile keine Dezimalzeichen gesetzt werden dürfen, wird dazu der Bruch verwendet. 56/10=5.6.

Ein Fixobjektiv 35mm mit durchgängig F2.8 würde so dargestellt: `35/1 35/1 28/10 28/10`

Ein guter Anhaltspunkt für die Schreibweise ist auch die Durchsicht der mit `exiftoolGUI` aus der Kameradatei ausgelesenen Tags.

Mehrere args_Dateien anlegen

Wenn Sie mit mehreren Objektiven Probleme haben, oder zu verschiedenen Objektiven die hinterlegten Daten noch individuell erweitern wollen, dann richten Sie mehrere Dateien im gleichen Aufbau mit den jeweils relevanten Daten ein.

Die Dateien werden in das Programmverzeichnis von `exiftoolGUI` gespeichert. Dann ist auch keine Pfadfestlegung erforderlich.

Inhalt der Datei „exiftoolGUI.ini“

Die Datei befindet sich ebenfalls im Programmverzeichnis von exiftoolGUI und kann mit dem Text-Editor geöffnet werden. Es ist angeraten, vor einer Änderung jeweils ein Backup zu erstellen.

Der folgende Ausschnitt zeigt das untere Ende der INI-Datei. Dort befindet sich das Kapitel [ETdirect] Bogdan Hrastnik hat bereits einige Einträge hinterlegt, die um die eigenen Einträge ergänzt und erweitert werden. Anhand des Aufbaus der vorhandenen Zeilen können eigene vordefinierte Aufgaben hinterlegt werden. Auch hier ist die genaue Schreibweise mit allen Zeichen und (fehlenden) Leerstellen zu beachten.

```
[TagList]
CustomView=-Exif:DateTimeOriginal -Makernotes:LensType -Exif:FNumber -Exif:ISO -Exif:FocalLength -
Makernotes:Quality -Exif:LensModel -Exif:DNGLensInfo -LensInfo -LensID -XMP:LensManufacturer<
MarkedTags=Model LensType<

[ETdirect]
Schreibe ObjDaten QUANTARAY 35-80mm=-if "$lensID# eq 25511" -@ QUANTARAY_35-80.args
Schreibe ObjDaten QUANTARAY 70-300mm=-if "$lensID# eq 25611" -@ QUANTARAY_70_300.args
Schreibe ObjDaten SIGMA 600-Mirror=-if "$FNumber# eq 1" -@ SIGMA_600_Mirror.args
Schreibe ObjDaten SOLIGOR MC 100=-if "$lensID# eq 136" -@ SOLIGOR_MC_100.args
Schreibe ObjDaten TAMRON 10-24mm WW=-if "$lensID# eq 255" -@ TAMRON_10_24mm_WW.args
Schreibe ObjDaten TAMRON 90mm MACRO=-if "$lensID# eq 28" -@ TAMRON_90mm_MACRO.args
```

Syntax der Einträge

Der folgende Auszug ist extra groß dargestellt, um das <Minuszeichen> vor dem „if“ hervorzuheben:

[Hinweistext zur Anzeige]=-if „,\$[Tag]# eq NNNN“ -@ [ARGS-Datei]

Das Kommando lautet in Normalsprache: *Wenn der Wert des eingesetzten Tags identisch ist mit dem Wert NNNN, dann verarbeitete den Inhalt der args-Datei.*

Bei Quantaray wird als Tag <LensID> verwendet, weil diese von der Kamera ausgelesen werden kann. Das SIGMA 600mm ist ein Uralt-Spiegeltele ohne Autofokus/Blendensteuerung mit Fixblende f8. Deshalb enthält es keinen Objektiv-Chip. Die Kamera erkennt das manuell nutzbare Objektiv nicht und gibt automatisch „Blende 1.0“ an. In dieser Option wird als Prüf-Tag <FNumber> gewählt: „wenn FNUMBER =1 dann“

In der SIGMA-600-ARGS-Datei stehen z.B. die Anweisungen:

```
-ExifIFD:UserComment=SIGMA 600mm F8 Mirror ohne Chip
-ExifIFD:LensInfo=600/1 600/1 8/1 8/1
-ExifIFD:LensModel=600mm F8 MirrorTele (SIGMA)
-Sony:LensType#=0
-xmp:LensManufacturer=SIGMA
-xmp:Lens=600mm F8 MirrorTele (SIGMA)
-xmp:LensID#=0
-xmp:LensInfo=600/1 600/1 8/1 8/1
-DNGLensInfo=600/1 600/1 8/1 8/1
```

Download und Installation

Exiftool-Link: <http://www.sno.phy.queensu.ca/~phil/exiftool/>

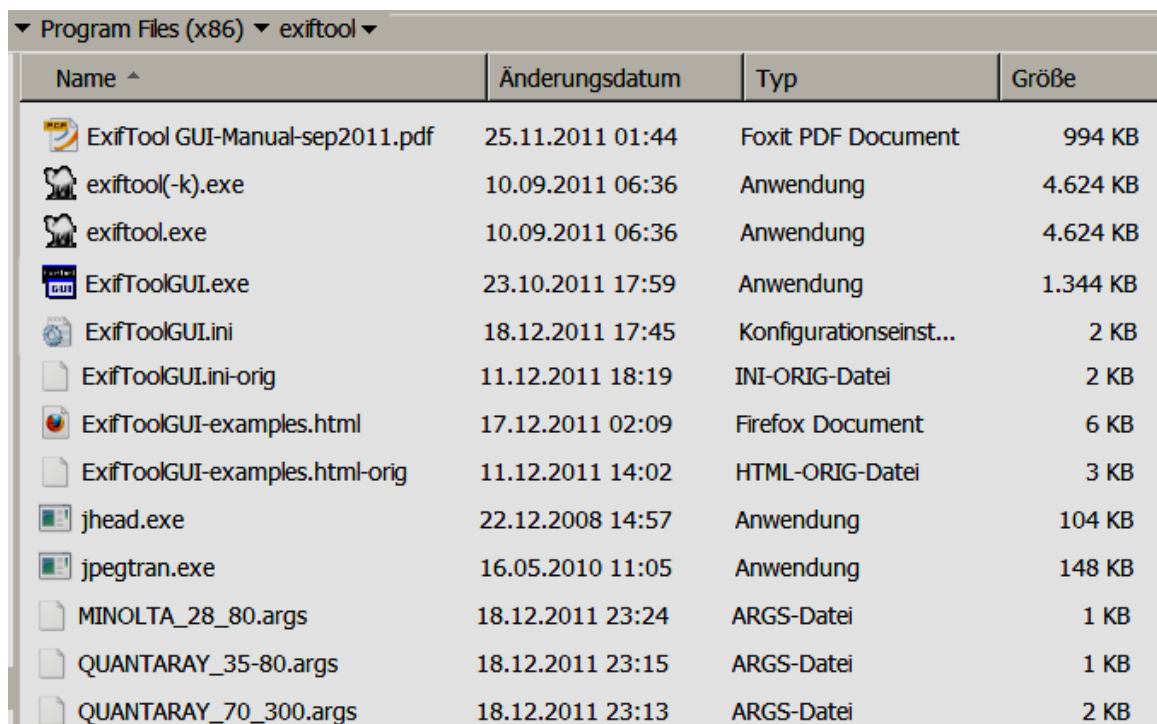
Das Paket kann von verschiedenen Webseiten heruntergeladen werden. Jedoch ist der direkte Weg zum Anbieter der sicherste und aktuellste. Das Programmpaket kommt als ZIP für verschiedene Betriebssysteme. Auf der gleichen Webseite ist weiter unten in der Rubrik „Related Utilities“ auch die **exiftool GUI** für Windows-Umgebung downloadbar.














Die Pakete kommen als ZIP und werden in ein beliebiges Verzeichnis entpackt. Das GUI-Paket sollte dann aber in das Programmverzeichnis integriert werden. Programmverzeichnis ist bei Windows-XP üblicherweise „C:\Programme\exiftool“. Bei Windows7 gibt es bei der 64-bit-Windows-Version das Verzeichnis C:\Programme (x86) für 32-bit-Programme. In diesem Fall wird das Paket in dieses Verzeichnis C:\Programme(x86)\exiftool kopiert.

Von der „exiftoolGUI.exe“ wird eine Verknüpfung auf dem Desktop angelegt. Damit hat man immer vom Desktop aus einen direkten Zugriff auf das Programm.

Zusätzlich wird die „exiftool(-k).exe“ in das Programmverzeichnis von exiftoolGUI kopiert. **Dann** wird von der exiftool(-k).exe eine Kopie erstellt und umbenannt in „exiftool.exe“

Und so sieht mein Programmverzeichnis mit Sicherungskopien und eigenen args-Dateien aus: (Ich habe mir zusätzlich das GUI-Manual als PDF heruntergeladen, und ich habe mir die „examples.html“ auf Deutsch übersetzt und um zusätzliche Beispiele erweitert. Die INI-Datei ist als Original-archiviert und die aktive INI-Datei enthält die oben beschriebenen Einträge.)



| Name ^ | Änderungsdatum | Typ | Größe |
|---|------------------|------------------------|----------|
|  ExifTool GUI-Manual-sep2011.pdf | 25.11.2011 01:44 | Foxit PDF Document | 994 KB |
|  exiftool(-k).exe | 10.09.2011 06:36 | Anwendung | 4.624 KB |
|  exiftool.exe | 10.09.2011 06:36 | Anwendung | 4.624 KB |
|  ExifToolGUI.exe | 23.10.2011 17:59 | Anwendung | 1.344 KB |
|  ExifToolGUI.ini | 18.12.2011 17:45 | Konfigurationseinst... | 2 KB |
|  ExifToolGUI.ini-orig | 11.12.2011 18:19 | INI-ORIG-Datei | 2 KB |
|  ExifToolGUI-examples.html | 17.12.2011 02:09 | Firefox Document | 6 KB |
|  ExifToolGUI-examples.html-orig | 11.12.2011 14:02 | HTML-ORIG-Datei | 3 KB |
|  jhead.exe | 22.12.2008 14:57 | Anwendung | 104 KB |
|  jpegtran.exe | 16.05.2010 11:05 | Anwendung | 148 KB |
|  MINOLTA_28_80.args | 18.12.2011 23:24 | ARGS-Datei | 1 KB |
|  QUANTARAY_35-80.args | 18.12.2011 23:15 | ARGS-Datei | 1 KB |
|  QUANTARAY_70_300.args | 18.12.2011 23:13 | ARGS-Datei | 2 KB |

Übrigens gibt es regelmäßig Updates der exiftool(-k).exe. Diese wird entpackt und wie oben beschrieben im Austausch als „exiftool(-k).exe“ und „exiftool.exe“ in das Verzeichnis kopiert. Die GUI wird seltener aktualisiert.

geo