

Dateiformate

An dieser Stelle erfolgen keine tieferschürfenden Betrachtungen zu irgendwelchen Klangunterschieden der Dateiformate oder den verschiedenen Auflösungen (Bitraten).

Um Musik auf unserem Speicher abzulegen, werden unterschiedliche Dateiformate zur Wahl gestellt. Wir können unsere vorhandenen CDs auslesen = rippen und dabei bestimmen, in welchem Dateiformat das erfolgen soll. Wir können aber auch in verschiedenen Internet-Portalen legal Musik käuflich erwerben und haben dabei oftmals die Möglichkeit, aus mehreren Formaten zu wählen.

Das Dateiformat erkennt man an der sogenannten Extension, der Endung des kompletten Dateinamens.

In der Musikwelt trifft man häufig auf mp3, wav, flac, wma, aac. Ein Dateiname kann z. B. so aussehen "yesterday.**wav**"

Welches Dateiformat ist "richtig"?

Das ist nicht zu sagen. Oftmals stehen spezielle Dateiformate im direkten Kontakt mit wirtschaftlichen Interessen, um Kunden an ein System zu binden. Seitens der Gerätehersteller und Downloadportale sind auch Lizenzgebühren oftmals ein Thema. So können z. B. wma-Dateien aus lizenzrechtlichen Gründen nicht mit der Software iTunes abgespielt werden. Da sich aber solche Lizenzgeschichten immer wieder ändern können, ist hier viel Bewegung in der Szene.

Bei den gekauften Musikdateien können weitere Einschränkungen durch Kopierschutz-Maßnahmen vorhanden sein.

Und natürlich spielt auch der Einsatzzweck eine große Rolle. Dateiformate beinhalten meistens Komprimierungsverfahren, was die Dateigrößen zum Teil drastisch reduziert. Auf kleinen MP3-Playern oder für Dauerberieselung im Auto kann das ein Argument sein.

Insgesamt bieten sich Formate an, die sehr weit verbreitet und mit möglichst vielen Geräten, Programmen und Systemen kompatibel sind. Bei der Gerätewahl sollte somit auch berücksichtigt werden, dass möglichst viele oder zumindest das bevorzugte Dateiformat abgespielt werden kann. Fast immer lassen sich die gebräuchlichsten Dateiformate auch konvertieren.

Komprimierte oder unkomprimierte Dateiformate?

Zum besseren Verständnis wird das Grundprinzip jetzt sehr oberflächlich beleuchtet.

Unser Rechner ist fast so doof wie ein Toastbrot. Er kann nur mit "0" und "1" rechnen. Das aber saumäßig schnell. Bevor wir "0" sagen können, kann der Rechner damit schon Millionen Rechenprozesse durchführen. Die kleinste Informationseinheit in einem Rechner ist ein Bit. Das ist entweder eine "1" oder eine "0". Aus 8 Bit setzt

sich ein Byte zusammen und ist somit die kleinste adressierbare Informationseinheit. Ein Byte sieht demnach aus: 00110010 oder 11000100.

Nehmen wir jetzt der Einfachheit halber einmal ein Foto. Wir knipsen einen kleinen Gegenstand vor einer großen, weißen Wand. In der Datei steckt jetzt für jeden Bildpunkt eine Information in Form eines Bytes.

Nehmen wir weiter an, dass unsere weiße Wand für jeden Bildpunkt den Wert 00010011 besitzt. Dann stehen praktisch jede Menge Bytes mit dem gleichen Wert hintereinander.

Da wir wissen, dass unser Byte immer 8 Bit haben muss, können wir die ersten "0" weglassen und erhalten: 10011. Um den Ursprungswert wieder zu erhalten, füllen wir einfach wieder die 3 fehlenden "0" auf und erhalten: 00010011.

Ein andere Variante wäre, dass wir die gleichen aufeinanderfolgenden Bytes unserer weißen Wand zählen. Aber anstatt, dass wir jetzt 00010011 hintereinander 100 mal aufschreiben, schreiben wir lieber "100 x 00010011" und können bei Bedarf diese Bytes ebenfalls wieder in der korrekten Menge zurückbekommen. Kommt unser "Zählwerk" an das kleine Motiv, sind abweichende Informationen (Bytes) vorhanden, die dann wieder extra analysiert werden.

Selbstverständlich können Verfahren auch kombiniert werden. In jedem Falle benötigen wir ein einen Komprimierungs- und den passenden Dekomprimierungs-Algorithmus.

Nehmen wir wieder unsere weiße Wand mit dem kleinen Objekt. Jetzt ist aber zusätzlich ein kleines Dreckkorn auf der Wand.

Nun müssen wir uns die Frage stellen, ab wann dieses Dreckkörnchen sichtbar wird. Dazu müssen wir erst abklären, wie hoch eigentlich die Auflösung unseres Monitors, ja vielleicht sogar des Auges, ist. Vielleicht kann diese Information wirklich weg, ohne dass es auffällt. Wie gucken also, wie groß der Fleck ist. Das heißt: Wie viele und wie weit abweichende Bytes vorhanden sind (gegenüber der weißen Wand) und schmeißen diese einfach weg. Allerdings wird ein kleines Detail unseres Motives genau so behandelt.

Auf Musik übertragen können wir das ungefähr vergleichen, als würde eine Mücke neben einem Presslufthammer vor sich hinsurren. Die Mücke kann weg, die hören wir sowieso nicht. Nach landläufiger Meinung kann der Mensch nur bis 20 000 Hz hören, also können in dem Falle alle Frequenzen, die darüber liegen, ebenfalls weg.

Sowohl bei unserem Bildchen als auch beim Ton spielt daher "wieviel kann weg, ohne dass es auffällt?" die alles entscheidende Frage. Daher richtet sich das Maß der Komprimierung (letztlich der Platzbedarf) danach, wie hoch wir "auflösen" wollen. Das ist bei einer Bilddatei (z.B. jpg) oder Tondatei (z.B. mp3) absolut vergleichbar.

Da wir aber die Information (egal ob Dreckkorn oder Mücke) wirklich weggeschmissen haben, bekommen wir diese Information auch nie wieder zurück.

Der Unterschied beider Komprimierungsansätze liegt auch in der Wiedergabe. Bei den Verfahren der "Byte-Berechnung" müssen wir vor Gebrauch der Daten diese erst entpacken, um eine 1:1 Reproduktion zu erhalten.

Wir drücken auf den "Play-Knopf" und während die Daten durch die verschiedenen Stufen unserer EDV flutschen, werden sie entpackt während wir hören. Das heißt natürlich, dass hier während der Nutzung die Recheneinheiten stärker belastet werden. Aber auch hier gibt es Unterschiede. Man kann die Tondatei "segmentweise" entpacken oder in einem Schwung in den Arbeitsspeicher legen. Da gibt's viele Möglichkeiten.

Bei den "wegrechnenden Komprimierungsverfahren" muss bei der Wiedergabe nichts entpackt werden, die Info ist sowieso weg. Hier muss der Abspieler nur das Format deuten können. Ist somit ein MP3 auf kleinste Speichergröße getrimmt (= nur geringe Auflösung), bleibt das Ergebnis immer Klangbrei, egal, wie schnell der Rechner oder wie clever die Abspielsofti ist. Und bei einem Bild bleibt es unscharf, selbst wenn ein 4-K-Monitor eingesetzt wird.

Daraus ergibt sich schon mal eine wichtige Erkenntnis für das Dateiformat unseres Musikarchivs: Es sollte sich in jedem Falle verlustfrei umrechnen lassen oder unkomprimiert vorliegen. Daher kommt z. B. MP3 als **Archiv**format grundsätzlich nicht in Frage.

WAV:

Dabei handelt es sich um ein sehr altes Format aus der Microsoft-Ecke. Auf Grund seines Alters **gab** es Einschränkungen bei zusätzlichen Informationen, den sogenannten ID-Tags. In diesen Infos werden Künstler, Titel, Album, Komponist usw. aufgeführt. Diese Einschränkung gilt aber nicht mehr. Anfang 2000 wurde das Format um die sogenannten Metadaten erweitert und wird als Broadcast Wave Format bezeichnet. In der Praxis blieb es bei der Datei-Endung "wav". Als Studioformat hat es einen sehr hohen Verbreitungsgrad. Oft liest man auch zum WAV-Format "PCM-Rohdaten". Dass das WAV-Format von der Breite des Publikums etwas stiefmütterlich behandelt wird, liegt wohl daran, dass viele der Meinung sind, dass die ID-Tags fehlen. Es ist also eher ein menschliches, als ein wirklich technisches Problem. Als unkomprimiertes Format erfordert es viel Speicherplatz, was vor einigen Jahren wegen der hohen Speicherpreise natürlich auch noch ein Negativ-Punkt war.

AIFF:

Ebenfalls ein unkomprimiertes Format, aber aus dem Apple-Lager. Entsprechend ist auch in den Studios der Verbreitungsgrad. Metadaten sind möglich. Da es unkomprimiert ist, bedeutet das natürlich wieder: hoher Speicherbedarf. Im Kern sind AIFF und WAF vergleichbar.

FLAC:

Ungefähr zur gleichen Zeit als das WAV-Format um die Metadaten erweitert wurde, erdachte die OpenSource-Ecke "FLAC". Dies ist ein freies Format ohne Lizenzgebühren, dafür mit Metadaten und verlustfreier Komprimierung. Im Vergleich zum WAV-Format benötigt FLAC ungefähr 50 % Speicherplatz und ist sehr weit verbreitet. Im HiFi-Heimbereich kann man eigentlich vom Standard reden.

ALAC:

Verlustfreie Komprimierung aus dem Apple-Lager. Der Code wurde aber vor kurzem freigelegt, so dass auch eine Zukunftssicherheit durch andere Hersteller (Verbreitungsgrad) bestehen dürfte. Wenn man so will, ist es das Gegenstück zu FLAC, hat aber noch nicht so den Verbreitungsgrad erreicht. Auch im Speicherbedarf ist es so ungefähr mit FLAC vergleichbar.

WMA:

Ursprünglich das verlustbehaftete Komprimierungsverfahren aus dem Windows-Lager. Später kam dann das verlustfreie WMA hinzu. Durch die Einbindung von DRM-Kopierschutz trifft man es in Kauf-Online-Portalen recht häufig an und es ist auch bei den Geräteherstellern relativ weit verbreitet.

MP3:

Ein sehr "altes" Format, u. a. vom Fraunhofer-Institut mitentwickelt. Nutzt hörpsychologische Effekte zur Komprimierung und ist somit verlustbehaftet, was aber in extremer Abhängigkeit zur Komprimierungsrate steht. Ursprünglich ohne Metadaten, die wurden erst später implementiert.

Trotz des "Alters" immer noch ein prima "Unterwegsformat", weil es unglaublich weit verbreitet ist und nahezu jeder Portable-Player mit diesem Format umgehen kann. Fast jede Computersoftware oder jeder Netzwerkplayer funktioniert problemlos mit MP3. Als dauerhaftes Archivformat ist es wegen des verlustbehafteten Komprimierungsverfahren nicht unbedingt erste Wahl.

AAC:

Ein relativ modernes, aber verlustbehaftetes Komprimierungsverfahren, was noch kleinere Dateien als MP3 ermöglicht, ohne dass es "schlechter klingt". Dadurch ist es in tragbaren Geräten sehr häufig anzutreffen. Mehrkanalton, Kopierschutz und Metadaten waren von Anfang an integriert.

Daneben gibt es auch noch eher "geschlossene Gesellschaften", wie z. B. das MLP-Format, u. a. von Meridian entwickelt und noch ungefähr 30 weitere Audioformate. Was sich auf Dauer als Standard etabliert, welches Format ausstirbt oder ob es noch weitere Formate geben wird, steht in den Sternen. Hinzu kommt natürlich, dass Heimanwender und Profisektor (z. B. Rundfunk) unterschiedliche Anforderungen stellen und es somit weiterhin Parallelwelten geben wird.