Der Desktop-Indexierer DocFetcher

Gut indexiert ist halb gefunden

Martin Loschwitz

DocFetcher erstellt einen Index, der eine schnelle Suche im Inhalt Ihrer Dateien erlaubt. Was das Programm leistet und wodurch es sich von den KDE-Bordmitteln unterscheidet, verraten wir in diesem Artikel.

Wenn Sie Ihren Computer regelmäßig nutzen, um Textdokumente – etwa für Ihre private Korrespondenz – zu verfassen, kennen Sie vielleicht das Problem: Mit der Zeit wächst der Berg vorhandener Dokumente auf Ihrer Festplatte, und es wird zusehends schwieriger, den Überblick zu behalten. Dabei sind Textdateien nicht die einzigen Dokumente, die zum Datenchaos beitragen – in den Home-Verzeichnissen der Anwender stapeln sich auch Grafikdateien, digitale Bücher (E-Books), Musikdateien, PDF-Dokumente und viele andere Dateitypen. Das führt zu schrägen Effekten: Viele Nutzer fangen von vorne und mit einem frischen Dokument an, wenn sie etwa ein Abo kündigen wollen – obwohl ähnliche Briefe bereits existieren, die man einfach kopieren und entsprechend anpassen könnte. Ebenso verhält es sich mit aus dem Netz heruntergeladenen PDF-Dateien: Die finden oft den Weg auf die Platte, weil man sie einmal gesucht und gefunden hat – später erinnert man sich dann oft nicht mehr daran, wo genau die Datei abgelegt ist und wie sie heißt. Die Folge: Viele Nutzer suchen sie im Netz erneut und laden sie ein weiteres Mal herunter. Im *Downloads*-Ordner findet sich oft dieselbe Datei etliche Male, zu erkennen am verräterischen, vom Browser beim Herunterladen angehängten Zahlensuffix (*Dokumentname (1).pdf*).

Wenn Ihnen dieses Problem bekannt vorkommt, gehören Sie zur Zielgruppe der Desktop-Indexierer (seltener auch Indizierer genannt). Der Ansatz dieser Programme ist nicht neu: Apple sorgte vor etlichen Jahren mit Spotlight in macOS erstmals dafür, dass diese Art von Programm Verbreitung fand. Desktop-Indexierer arbeiten anders als ein klassisches Suchwerkzeug wie *locate*, das bis vor wenigen Jahren auf Linux-Systemen der Standard für die Suche war.

Indexierer durchforsten die Platte

locate ist ein Beispiel für einen sehr simplen Suchdienst: Das Tool legt eine Liste aller Dateien des Systems an, die es durchforstet, wenn der Nutzer auf der Kommandozeile via

```
locate Name
```

eine Datei sucht. Dabei ist das einzige Kriterium, das *locate* bei der Suche berücksichtigt, der Dateiname: Den Inhalt der Dateien kennt *locate* nicht, so dass es unmöglich ist, damit nach Inhalten zu suchen. Hier kommen Tools wie das Programm DocFetcher [1] ins Spiel, das Sie auch auf der Heft-DVD finden: Die durchforsten in regelmäßigen Abständen Ihre Festplatte und legen eine lokale Datenbank mit Informationen zu den vorhandenen Dateien und deren Inhalten an. Suchen Sie dann nach einer bestimmten Zeichenkette, liefert DocFetcher Ihnen nicht nur die Dateien, deren Dateiname mit der Suchzeichenkette übereinstimmt, sondern auch Dateien, in deren Inhalt der gesuchte Begriff vorkommt.

Das ist viel effektiver als die Suche auf Basis der Dateinamen. Wenn Sie Ihren Computer auch nur einigermaßen regelmäßig verwenden, sind Werkzeuge wie DocFetcher effektiv die einzige Variante, um nicht den Überblick zu verlieren.

Wie Indexierer funktionieren

Der Vergleich von Indexierern mit *locate* hinkt ein wenig: Dass Werkzeuge, die Dateien rein auf Basis ihres Dateinamens finden, nur eingeschränkt nützlich sind, haben Softwareentwickler bereits vor Jahren erkannt. Bald kamen erste Tools auf den Markt, die nicht nur die Namen von Dateien, sondern auch deren Inhalte untersuchen konnten. Das Prinzip ist simpel: Sobald Sie einen Begriff in das Suchfeld eingeben, läuft das Tool los und durchforstet den gesamten Inhalt Ihrer Festplatte oder einen Teil von dieser, etwa Ihr persönliches Verzeichnis. Besonders effizient ist das aber auch nicht, und obwohl moderne Desktopsysteme und Laptops heute oft mit sehr schnellem Flash-Speicher ausgestattet sind, dauert eine solche Suche lange.

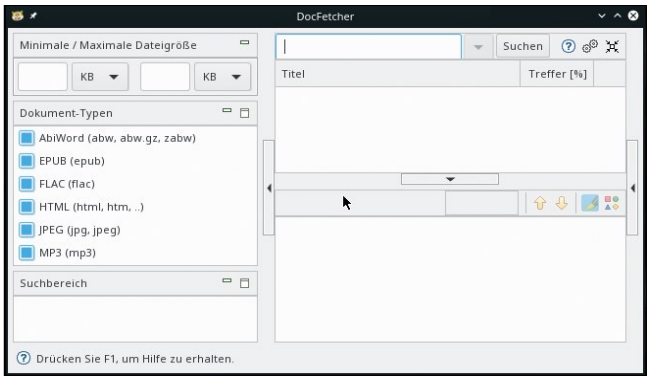

Abb. 1: So sieht DocFetcher direkt nach der Installation aus: Es fehlen noch die Verzeichnisse, die DocFetcher indexieren soll.

Desktop-Indexierer erweitern das Konzept um eine Datenbank, in der sie als Ergebnis eines einmaligen Scans aller unterstützten Dokumente deren Inhalte speichern. Damit das System nützlich bleibt, aktualisieren sie diesen Index zudem regelmäßig. Tippen Sie nun einen Begriff in das Suchfeld ein, sucht das Programm nicht mehr in den Dateien auf Ihrer Festplatte, sondern nur noch in seiner eigenen Datenbank. Das Einlesen der gesamten Festplatte geschieht nur, wenn der Indexierer seine Datenbank anlegt oder aktualisiert (und dann auch nur für neue oder geänderte Dateien), jedoch nicht mehr bei jeder Suche.

Mittlerweile gibt es mehrere Werkzeuge, die versprechen, die Indexierung der Festplatte unter Linux zuverlässig, gut und schnell zu erledigen. Alle gängigen Distributionen bringen einen entsprechenden Dienst mit; bei KDE heißt er Baloo [2]. DocFetcher ist eine sehr bekannte, desktopunabhängige Alternative.

Wir stellen DocFetcher im Detail vor, beschreiben seine besonderen Funktionen und gehen auf die Unterschiede zum KDE-Tool Baloo ein, das sehr ähnliche Ziele verfolgt.

Simpel dank Java

Sie finden DocFetcher auf der Heft-DVD – nicht als Distributionspaket in einem der gängigen Formate *.rpm* oder *.deb*, sondern in Form einer *.zip*-Datei. Wenn Sie die *.zip*-Datei auf Ihr Linux-System kopiert haben, entpacken Sie sie mit *ark* in Ihrem Home-Verzeichnis und starten das Java-Programm DocFetcher anschließend per Klick auf *DocFetcher-GTK3.sh*. KDE wird Ihnen anbieten, die Datei zu öffnen oder auszuführen – bei diesem Dialog wählen Sie *Ausführen*. Kurze Zeit später erscheint das GUI von DocFetcher auf dem Bildschirm (**Abbildung 1**).

Da es Java-Versionen für Windows, macOS und Linux gibt, läuft DocFetcher auf all diesen Betriebssystemen: Solange eine Java-Laufzeitumgebung vorhanden ist, funktioniert das Programm vollkommen problemlos. Lediglich die Startmethode unterscheidet sich: Wer DocFetcher unter Windows nutzen möchte, klickt dort auf *DocFetcher.exe*. Unter macOS kommt das Application Bundle *DocFetcher.app* zum Einsatz und unter Linux eben das schon erwähnte Shell-Skript *DocFetcher-GTK3.sh*. Das GUI sieht auf allen Systemen gleich aus.

Aus der Plattformunabhängigkeit ergibt sich ein interessantes Anwendungsszenario: Formatieren Sie z. B. einen USB-Stick mit dem FAT32-Dateisystem, ist er unter Linux, macOS und Windows les- und schreibbar. Wenn er Ihre private Briefkorrespondenz enthält, können Sie von allen Betriebssystemen aus darauf zugreifen. Wenn Sie DocFetcher in einem Verzeichnis auf dem USB-Stick ablegen und es nutzen, um in diesem Ordner den Inhalt des USB-Sticks zu indexieren, haben Sie einen portablen Dokumentenspeicher mit eingebauter Suchmaschine. Egal, ob Sie diesen an einen Computer mit Windows, Linux oder macOS anschließen, sobald Sie dort die jeweilige DocFetcher-Version starten, können Sie den Index für schnelle Suchen nutzen. Auch Index-Updates können Sie jederzeit (unabhängig vom gerade laufenden System) anstoßen.

Das DocFetcher-GUI

Nach der DocFetcher-Installation und dem ersten Start des Programms zeigt dieses sein Hauptfenster (**Abbildung 1**). Das ist in vier Bereiche unterteilt: Links oben ist eine Auswahlliste der Dateitypen aufgelistet, innerhalb derer eine Suche beim

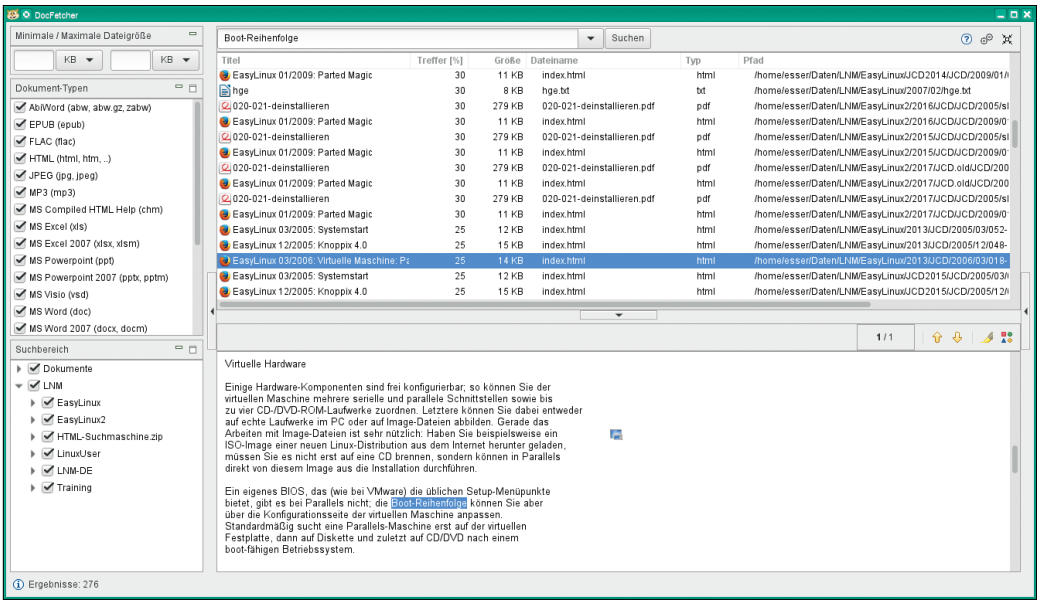

Abb. 2: Eine Suche bringt nach wenigen Sekunden eine Vielzahl von Ergebnissen, die DocFetcher nach Relevanz sortiert. In der Textvorschau wird der gefundene Suchbegriff hervorgehoben.

nächsten Mal stattfindet. Indem Sie hier Häkchen vor Einträge setzen oder entfernen, schließen Sie die entsprechenden Dateitypen von der Suche aus oder ein. Direkt darüber geben Sie zudem an, ob bei der Suche Dateien eingeschlossen sein sollen, die eine bestimmte Größe über- oder unterschreiten.

Darunter (unten links) können Sie Pfade angeben, die DocFetcher bei seiner Suche einschließen oder auslassen soll. Das kann dann besonders sinnvoll sein, wenn Sie sehr viele Dateien auf der Platte und damit auch einen großen Such-Index haben. Denn dann kann sogar die schnelle DocFetcher-Suche viel Zeit benötigen. Schließen Sie unten links einen Teil Ihres Dateisystems mit vielen Dateien aus, wird die Suche deutlich schneller.

Der rechte Teil des DocFetcher-Fensters ist der Suche gewidmet. Oben haben Sie zunächst ein Eingabefeld – hier geben Sie einen beliebigen Begriff an, nach dem Sie in Ihrem Index suchen möchten. Die Anzeige ist ausgefeilt: Bei den Resultaten gibt DocFetcher links den *Titel* an, den eine Datei trägt, in der es den Suchbegriff gefunden hat. Haben Sie etwa eine lokale Kopie eines Wikipedia-Artikels im HTML-Format auf der Platte, in dem DocFetcher bei einer Suche den gewünschten Begriff findet, würde es in diesem Feld den Titel der Wikipedia-Seite anzeigen. Ähnliches gilt für Textdokumente. Weil DocFetcher diese komplett analysiert, legt es die entsprechenden Informationen ebenfalls in seiner Datenbank ab.

Weitere Felder in der Anzeige der Suchergebnisse sind nützlich: Als Prozentwert gibt DocFetcher an, wie hoch die Wahrscheinlichkeit ist, dass ein Ergebnis in der Liste genau das enthält, was Sie suchen. Haben Sie etwa ein LibreOffice-Dokument mit der Betreffzeile „Kündigung meines Abos“ auf Ihrer Platte und suchen dann nach „Kündigung meines Abos“ in DocFetcher, zeigt das Programm das LibreOffice-Dokument mit einem hohen Prozentwert an. Dateien, in denen zwar die Worte „Kündigung“, „meines“ und „Abos“ vorkommen, jedoch nicht als zusammenhängende Zeichenkette, erhalten einen niedrigeren Prozentwert (**Abbildung 2**).

Der größte Teil des Fensters unten rechts schließlich zeigt eine Vorschau des Suchergebnisses, das Sie in der Trefferliste oben auswählen. Bei HTML-Dokumenten lässt sich zwischen einer reinen Textvorschau und der Darstellung wie im Browser wählen, wobei im Browsermodus die Treffer nicht angezeigt werden.

Ordner bestimmen

DocFetcher folgt dem Prinzip, dass es nur relevante Teile der Festplatte in den Index aufnimmt. Dazu ein Beispiel: Wenn Sie Werkzeuge wie VirtualBox oder VMware nutzen, um auf Ihrem System virtuelle Systeme zu betreiben, liegen deren virtuelle Festplatten vielleicht im Ordner *VMs* in Ihrem Home-Verzeichnis. Je nach Art und Zustand einer solchen VM kann ein Platten-Image etliche Gigabyte Speicherplatz belegen. Würde DocFetcher ab Werk einfach alle Unterverzeichnisse Ihres Home-Verzeichnisses indexieren, fiele auch jenes mit den VMs darunter. Die Analyse dieser Dateien würde lange dauern, wäre aber zwecklos: Auf Dateien innerhalb der VMs können Sie aus Ihrem Hauptsystem heraus ohnehin nicht direkt zugreifen.

Welche Ordner DocFetcher untersuchen soll, legen Sie in den Einstellungen fest. Dazu klicken Sie im unteren linken Teil *Suchbereich* des DocFetcher-Fensters mit der rechten Maustaste einmal in den weißen Bereich und wählen im erscheinenden Kontextmenü den Eintrag *Index erstellen aus / Ordner* aus. Danach öffnet sich ein Auswahldialog, in dem Sie das Verzeichnis angeben, das Sie indexieren möchten.

Am Anfang der DocFetcher-Nutzung steht also Konfigurationsarbeit. Haben Sie sich für einen Ordner entschieden, zeigt DocFetcher Ihnen noch ein weiteres Fenster an, in dem Sie festlegen, welche Dateitypen DocFetcher untersuchen soll. Das Fenster unterteilt sich in zwei Bereiche: Oben gibt DocFetcher Dateiendungen an, die zu Formaten gehören, die DocFetcher unterstützt. Darunter haben Sie auch die Möglichkeit, über so genannte reguläre Ausdrücke anzugeben, welche Dateien im Index enthalten oder von diesem ausge-

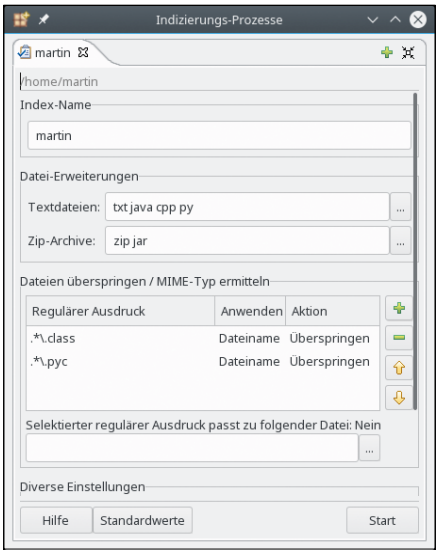

Abb. 3: Welche Dateien DocFetcher beim Anlegen des Indexes beachten soll, legen Sie in den Einstellungen fest.

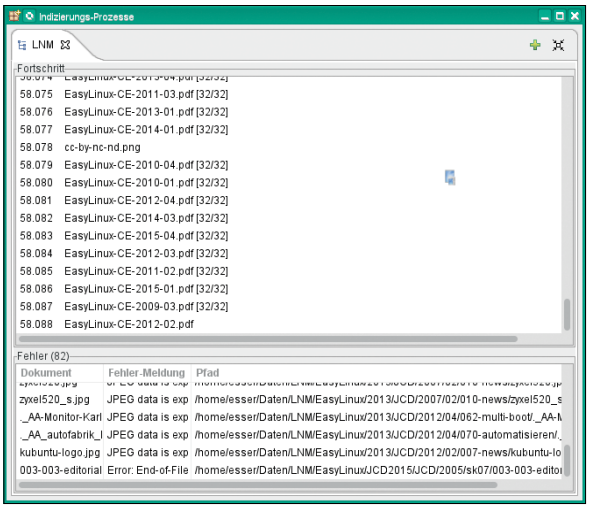

Abb. 4: Ein sehr großes Dokumentenverzeichnis erstmals zu indexieren, kann durchaus Stunden dauern.

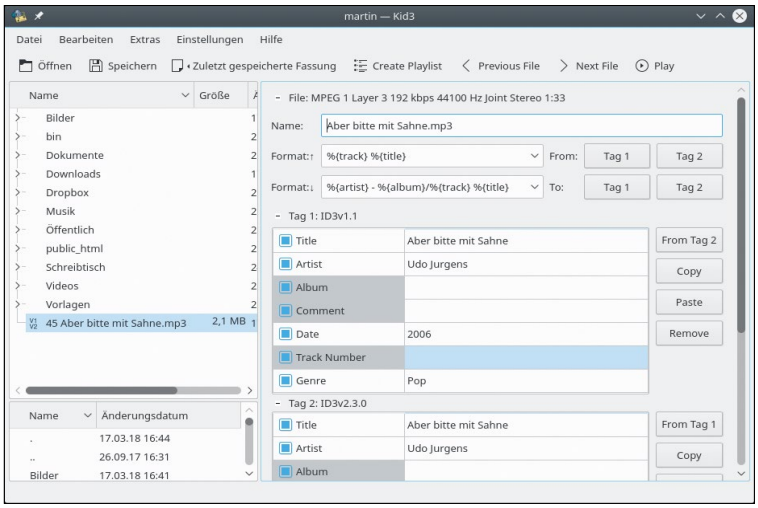

Abb. 6: MP3-Dateien enthalten ID3-Tags, die etwa Titel und Künstler festlegen – die Abbildung zeigt den ID3-Tag-Editor Kid3.

schlossen sein sollen (Abbildung 3). Der Eintrag *.*\.vmwarevm* mit der Aktion *Überspringen* würde etwa dazu führen, dass DocFetcher virtuelle Maschinen von VMware explizit von der Suche ausschließt. Für VirtualBox-VM-Container mit Dateiendung *.ova* erreichen Sie denselben Effekt, indem Sie *.*\.ova* als Filter mit der Aktion *Überspringen* angeben. (Der Backslash \ vor dem Punkt ist jeweils nötig, weil in regulären Ausdrücken ein Punkt für ein beliebiges Zeichen steht; um nach Dateinamen mit einem Punkt vor der Endung zu suchen, muss dieser Punkt als \. angegeben werden.) Bei der Auswahl der Ordner, die DocFetcher für Sie analysieren soll, ist Augenmaß nötig, um genau die richtigen Verzeichnisse und Dateien in den Index aufzunehmen.

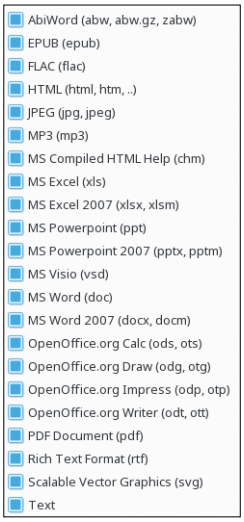

Abb. 5: DocFetcher versteht eine Vielzahl aktueller Dateiformate.

Ein Klick auf *Start* setzt die Indexierung in Gang (Abbildung 4). Wenn sich im angegebenen Ordner viele Dateien befinden, nimmt das erste Mal einige Zeit in Anspruch. Das gilt besonders dann, wenn Sie eine klassische Festplatte und keine schnelle SSD verwenden. Wollen Sie den Index später aktualisieren, genügt es, auf den jeweiligen Eintrag im *Suchbereich*-Fenster zu klicken und dort *Aktualisieren* auszuwählen.

Unterstützte Dateitypen

Eine der wichtigsten Eigenschaften eines Desktop-Indexierers ist die Liste der unterstützten Dateitypen. Nur Dateien, deren Format DocFetcher versteht, kann es in den Index aufnehmen. Mit den Microsoft-Office-Formaten (*.doc*, *.xls* und *.ppt*) und deren XML-basierten Nachfolgern (*.docx*, *.xlsx*, *.pptx*) kommt DocFetcher ebenso zurecht wie mit den freien Alternativen von LibreOffice (*.odt*, *.ods* sowie *.odp*). Digitale Bücher im EPUB-Format, HTML- und Textdateien sowie RTF-Dokumente (Rich Text Format) kann DocFetcher ebenfalls analysieren. Auch PDF-Dateien bereiten dem Tool keine Probleme (Abbildung 5).

DocFetcher beherrscht nicht nur die gängigen Office-Dokument-Formate, sondern versteht sich auch auf den Umgang mit anderen Dateitypen. Bei seiner Suche schaut sich das Programm etwa auch MP3-Dateien an. Zwar kann es hinterher nicht Lieder anhand einzelner Textzeilen finden, die darin vorkommen, nimmt aber die ID3-Tags (Abbildung 6) der Dateien mit in den Index auf. Wer also in DocFetcher nach einem Song- oder Albumtitel sucht, findet MP3-Dateien mit passenden Einträgen im ID3-Tag (Abbildung 7). Dasselbe gilt für FLAC-Dateien.

Zusätzlich beherrscht DocFetcher auch den Umgang mit Grafikdateien: Es wertet enthaltene EXIF-Tags aus und nimmt diese in den Index auf. Suchen Sie Fotos auf Ihrer Festplatte, die Sie mit einer bestimmten Blendeneinstellung oder an einem bestimmten Ort aufgenommen haben, wird DocFetcher sie in den Suchergebnissen anzeigen.

Hinzu kommt Unterstützung für Vektorgrafiken im SVG-Format und schematische Zeichnungen etwa aus Microsoft Visio. Beide Formate eint, dass in ihnen Text vorkommen kann – wenn Sie etwa einen Schaltplan als SVG-Datei gespeichert haben und diesen nun suchen, genügt es, ein Wort aus dem Schaltplan anzugeben, und schon fördert DocFetcher die entsprechenden Dateien zu Tage.

Mächtige Suchfunktion

Besonders stolz sind die DocFetcher-Entwickler auf den Suchdialog. Die einfachste Option, ihn zu nutzen, ist, einfach eine beliebige Zeichenkette einzugeben und danach durch die Trefferliste zu scrollen. Diese Form der Nutzung bleibt allerdings weit hinter den Fähigkeiten zu-

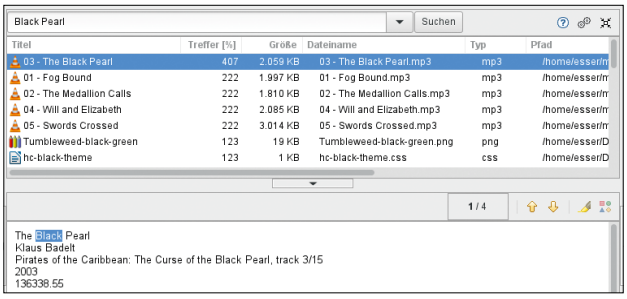

Abb. 7: Wer seine MP3-Sammlung mit indexiert, kann in DocFetcher auch nach Interpret, Album oder Titel suchen.

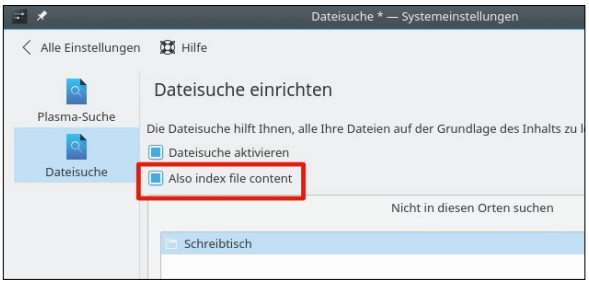

Abb. 8: Die KDE-Suche Baloo lässt sich über die KDE-Systemeinstellungen konfigurieren.

rück, die das unscheinbare Suchfeld verbirgt. Es unterstützt auch boolesche Operatoren, allerdings nur in englischer Sprache. Die bekanntesten Operatoren sind *OR* (oder), *AND* (und) und *NOT* (nicht). Eine Suche mit diesen Operatoren könnte etwa sein: *„Aber bitte mit Sahne“ OR „Vielen Dank für die Blumen“*.

Eine vollständige Übersicht in deutscher Sprache über die Möglichkeiten des Suchfelds erhalten Sie, indem Sie auf das Icon mit dem Fragezeichen neben der Suchleiste und dann auf *Suchanfrage-Syntax* im unteren Teil der geöffneten Hilfe klicken.

Wo DocFetcher Baloo aussticht

Am Ende der DocFetcher-Vorstellung stellt sich die Frage, warum Sie sich als Nutzer mit dem Programm überhaupt beschäftigen sollen – es gibt in KDE schließlich mit Baloo einen Dienst, der ganz ähnliche Funktionen bietet oder sie zumindest verspricht. Und anders als DocFetcher gehört Baloo fest zu einer KDE-Standardinstallation. Richten Sie also einen KDE-Desktop ein, ist Baloo schon mit dabei. Theoretisch ist in Sachen Datei-Indexierung bei KDE also alles in bester Ordnung.

Praktisch sieht es anders aus. Bei Google etwa türmen sich die Suchanfragen, wie man Baloo abschaltet. Denn viele Nutzer und gerade jene mit langsameren Festplatten sind von dem Dienst vorrangig genervt. Baloo indexiert grundsätzlich das gesamte persönliche Verzeichnis des Nutzers, wenn dieser die Einstellungen nicht explizit ändert. Wenn der Indexierer von Baloo losläuft, um den Index zu aktualisieren (und das tut es ab Werk in regelmäßigen Abständen), leistet die Festplatte Schwerstarbeit. Auch über die CPU-Last, die der Dienst hervorruft, gibt es viele Beschwerden in den KDE-Nutzerforen.

Immerhin: Baloo bietet mittlerweile ein funktionierendes Modul für die KDE-Systemeinstellungen. Hinter dem Eintrag *Suchen* verstecken sich dort die zu Baloo gehörenden Parameter. Vorsicht: Das Untermodul *Plasma-Suche* hat mit Baloo nichts zu tun – es gehört zu KRunner, einem Programm, das diverse KDE-eigene Dienste indexiert, etwa die vorhandenen Miniprogramme. Wollen Sie die Baloo-Konfiguration aufrufen, klicken Sie links auf *Dateisuche*.

Dass man beim KDE-Projekt dem eigenen Suchwerkzeug nur bedingt vertraut, wird nicht zuletzt dadurch deutlich, dass die Analyse der Dateinhalte (*Also index file content*) ab Werk deaktiviert ist (**Abbildung 8**).

Will man Ordner oder Dateien aus dem Baloo-Index entfernen, geht das zwar per KDE-Kontrollzentrum oder alternativ über eine Konfigurationsdatei. Dazu öffnen Sie in einem Texteditor *~/.config/baloofilerc*. Leider ist das die einzige Konfigurationsoption, die das GUI unterstützt. Andere Baloo-Einstellungen lassen sich also nur über die Konfigurationsdatei vornehmen, was nicht annähernd so komfortabel ist wie bei DocFetcher – wo Sie sämtliche Einstellungen per GUI erreichen.

… und wo nicht

Einen großen Vorteil hat Baloo gegenüber DocFetcher jedoch: Das Programm ist nahtlos in die KDE-Oberfläche integriert. Drücken Sie also im KDE-Dateimanager Dolphin [Strg-F], um ein Suchfeld zu öffnen, erhalten Sie unmittelbaren Zugriff auf die Baloo-Suche (**Abbildung 9**). DocFetcher hingegen ist ein externes Zusatzprogramm, das Sie als solches auch explizit starten müssen und das in den KDE-Desktop nicht integriert ist.

Fazit: Klein, aber oho

Die Arbeit mit DocFetcher macht gerade dann Spaß, wenn man auf einem nicht ganz aktuellen System mit wenigen Ressourcen und langsamer Festplatte unterwegs ist. Denn hier spielt das Tool seine Stärken voll aus: Es sucht nur dort nach Inhalten, wo Sie es explizit vorgeben – alle anderen Bereiche der Festplatte lässt DocFetcher in Ruhe. Dass das DocFetcher-GUI intuitiv nutzbar ist und die DocFetcher-Konfiguration leicht fällt, macht DocFetcher zu einem empfehlenswerten Suchprogramm.

Attraktiv ist der DocFetcher-Einsatz auch, wenn Sie USB-Sticks indexieren wollen, denn das Programm läuft auf allen gängigen Betriebssystemen. Sie können DocFetcher also einfach mit den zu indexierenden Dateien auf den USB-Stick kopieren – fertig. Das haben wir auch für den Index der EasyLinux-Archiv-DVD genutzt. (hge) ■

INFOS

[1] DocFetcher: *http://docfetcher.sourceforge.net/de/index.html* (*http://ezlx.de/k2g1*)

[2] Baloo: *https://community.kde.org/Baloo* (*http://ezlx.de/k2g2*)

SOFTWARE AUF DVD:

DocFetcher 1.1.19

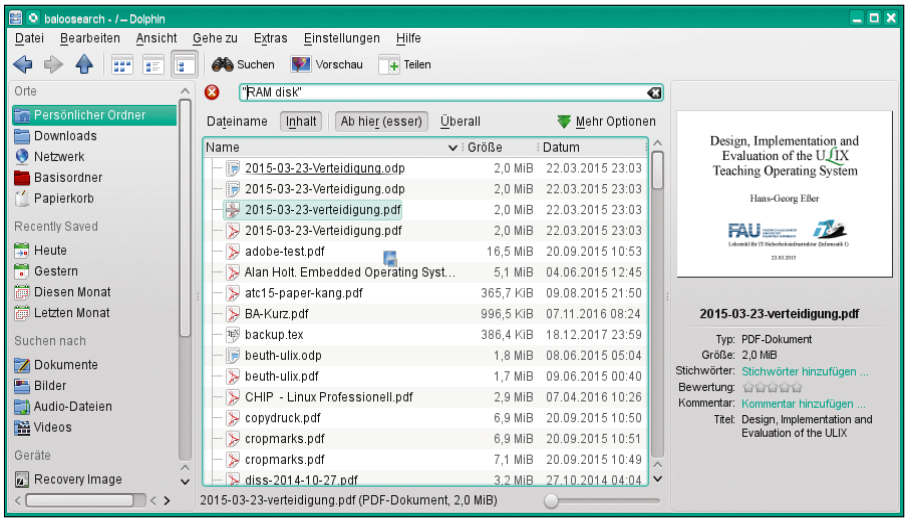

Abb. 9: Anders als DocFetcher ist Baloo in KDE integriert und lässt sich aus Dolphin heraus aktivieren – es ist kein Start einer zusätzlichen Anwendung nötig.