

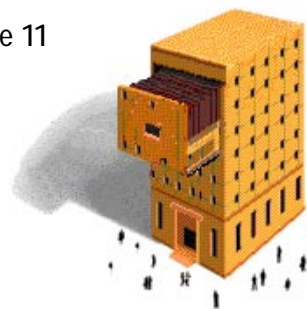
WORKSHOP



FileMaker 3.0

Folge 11

AppleScript



Die Scripting- und Formelfunktionen von FileMaker 3.0 lassen zwar kaum noch Wünsche offen, doch wenn Ihnen diese bei speziellen Datenbankproblemen nicht mehr weiterhelfen, bietet Ihnen AppleScript oder auch Frontier von Userland möglicherweise einen Weg aus der Sackgasse. Wollen Sie mit FileMaker zum Beispiel stilisierten Text importieren, Aiff-Sounds abspielen, Faxe und E-Mails direkt aus der Datenbank verschicken oder vollautomatisch Kataloge für Print, WWW und CD produzieren, dann führt an AppleScript kein Weg vorbei.

In den letzten Monaten war es verächtlich ruhig geworden um Apples Open Scripting Architecture (OSA), derer sich Programmentwickler be-

dienen müssen, um ihre Software für AppleScript tauglich zu machen. Doch seit dem 13. November widmet Apple der Technologie wieder die verdiente Aufmerksamkeit. Unter www.applescript.fovista.com oder www.applescript.apple.com gibt es auf einer umfangreichen Web-Seite endlich klare Perspektiven und viel Material für die von einigen bereits totgesagte OSA.

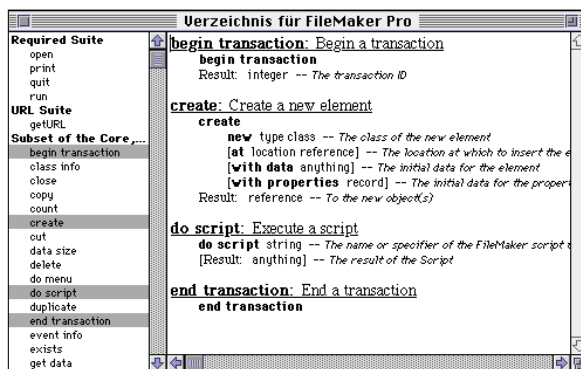
AppleScript ist schnell erlernt, wird von den meisten Applikationen unterstützt und sorgt für umfassende Erleichterungen in unterschiedlichen Arbeitsumgebungen. Einziger Wermutstropfen: Auf Power-Macs läuft AppleScript nur im emulierten Modus. Wer hohe Geschwindigkeit und somit eine leistungsfähigere Al-

ternative braucht, findet in dem oben bereits erwähnten Frontier die geeignetere Programmierungsumgebung.

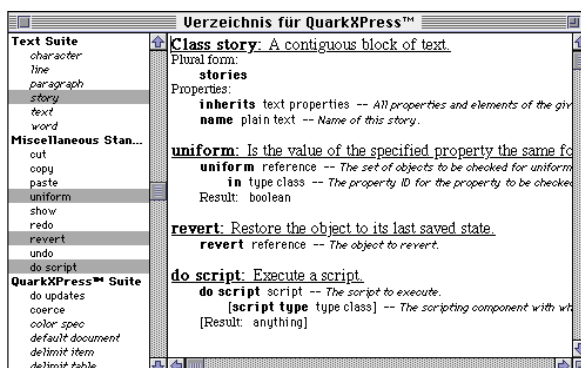
Traumpartner. Egal, ob die Hauptanwendung Text-, Bildbearbeitungs- oder Layoutaufgaben zu bewältigen hat – es gibt fast immer einen Grund, die anfallenden Daten mit einem ordentlichen Datenbankprogramm wie eben FileMaker zu verwalten. Leider unterstützen nicht alle Programme die Kommunikation über Apple Events/AppleScript, doch einige der wichtigen gehören dazu: Der DTP-Klassiker XPress etwa wartet mit einer gigantischen Bibliothek auf und verlangt geradezu nach einer intensiven Zusammenarbeit mit FileMaker. ➤

FileMaker-Workshop

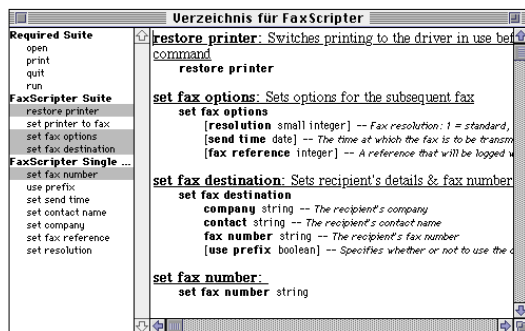
Mit Version 3 hat die im Mac-Bereich weitverbreitete Datenbank FileMaker den Schritt zur Relationalität gemacht. Was das konkret bedeutet und wie Sie die neuen Funktionen sinnvoll einsetzen können, zeigt Ihnen unsere Serie.



1 Welche Events ein Programm versteht, lässt sich leicht herausfinden, indem man im „Ablage“-Menü des Skripteditors unter „Verzeichnis öffnen ...“ das jeweilige Programm auswählt. Das Verzeichnis von FileMaker lässt kaum Wünsche offen.

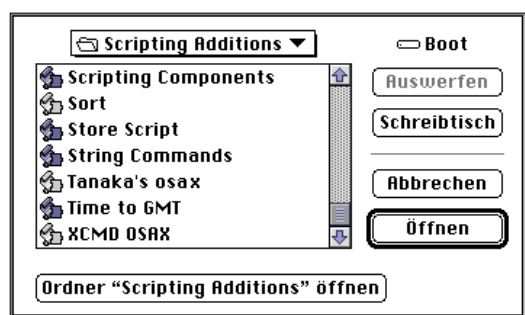


2 Das Verzeichnis von XPress ist geradezu gigantisch. Während sich in FileMaker schon nach wenigen Stunden eine grobe Übersicht gewinnen lässt, findet man hier eine Herausforderung für Wochen.

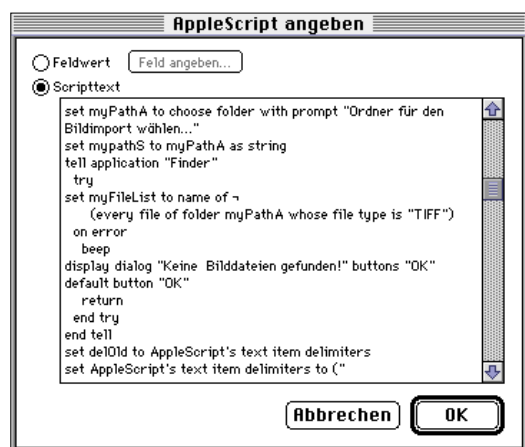


3 Die zur Zeit einfachste Faxanbindung über AppleScript bietet das Programm FaxExpress an.

FaxSTF liefert in seinem Developer Kit zwar keine direkte AppleScript-Anbindung, die mitgelieferten HyperCard-XCMDs lassen sich jedoch über eine geeignete Shell (zum Beispiel XCMD Shell) auch zur Kommunikation mit FaxSTF überreden. Der schnellere Weg allerdings ist der Erwerb des Programms FileFax.



4 Die Erweiterungen (Osaxen) im Ordner „Scripting Additions“ erweitern nicht nur die Möglichkeiten. Viele Prozeduren, die AppleScript nur in Schleifen („repeat with ...“) oder umständlichen Konvertierungen ausführen kann, lassen sich mit der richtigen Erweiterung um das Zifache beschleunigen, beispielsweise mit „ACME Script Widgets“ oder „Tanaka's osax“.



5 Der AppleScript-Editor in FileMaker eignet sich allerhöchstens zum Einsetzen fertiger Skripts oder zum Bearbeiten knapper Dreizeiler. Zum Editieren und Testen eines Skripts sollte man in jedem Fall auf Apples Skripteditor zurückgreifen.

Wer FileMaker-Anbindung in Bereichen wie Konvertierung, Medienintegration und Multimedia sucht, kann mTropolis, HyperCard, Cumulus, VideoFusion, SoundApp, JpegView oder clip2gif kontaktieren. Für Büroaufgaben eignen sich RagTime, WordPerfect, MS-Office sowie die Claris-Programme Works und Impact. Im Netz- und Kommunikationssektor offerieren Eudora, Navigator, Anarchie, Fetch, FaxExpress, FaxSTF (über XCMDs) und verschiedene WWW-Server Scripting-Möglichkeiten. An Utilities bieten StuffIt, ZipIt, ShrinkWrap, UltraFind und HTML-Markup ihre Dienste an. Für die Verwaltung des Schreibtischs stellt der Finder alle gewünschten Werkzeuge zur Verfügung. Eine vollständige Aufzählung aller skriptfähigen Programme würde natürlich den Rahmen dieses Artikels sprengen.

Event-Muffel. Macromedia hingegen setzt ebenso wie die Firma ACI ganz auf Plattformunabhängigkeit. So unterstützt Director nicht einmal die Events der „Required Suite“, also die von Apple erwartete Grundausstattung eines Programms. Das ist besonders deshalb ärgerlich, weil der Applikation brauchbare Datenbankfunktionen ebenso fehlen wie ein praktikabler Textaustausch über die Zwischenablage.

Auch Photoshop 3.0.5 und Premiere 4.2 lassen sich (noch) nicht über AppleScript fernsteuern. Ferner fehlen fast allen „Light Editions“ und „Limited Editions“ die benötigten Skriptfähigkeiten. Wer etwa die – unter anderem mit der Apple Workgroup Server Solution gelieferte – Faxsoftware 4Sight Fax per AppleScript in seine FileMaker-Datenbank einbinden will, muß sofort zum teuren Vollprodukt greifen.

Schnelleinstieg. Die Einsatzbereiche für das Duo AppleScript und FileMaker lassen sich grob dreiteilen: Das Einlesen von Daten (beispielsweise Bilder), die Übergabe von Daten (wie etwa Auswertungen) und periodisch wiederkehrende Aktivitäten. Die Arbeit mit AppleScript setzt natürlich die vollständige Installation sowie die Aktivierung der entsprechenden Sy-

stemerweiterung voraus. Der Ordner „FileMaker & Apple Events“, der sich auf der US-Version der FileMaker-CD-Extras oder auch unter der Web-Adresse www.claris.com/support/products/filemakerpro/docs/cd-extra.html (siehe auch MACup 12/96, Seite 194) befindet, erleichtert den Einstieg in das FileMaker-typische Scripting; zudem lassen sich viele der Skripts nach geringfügigen Änderungen sofort sinnvoll einsetzen.

Der richtige Aufenthaltsort. Die Möglichkeiten, ein (AppleScript-) Skript zu schreiben und aus FileMaker aufzurufen, sind vielfältig: ■ AppleScripts lassen sich in FileMakers Skripteditor ScriptMaker einlesen und editieren. Leider sind hier keine Operationen erlaubt, die Bezug auf FileMaker und Objekte der Datenbank nehmen. Fehlende Auszeichnungsmöglichkeiten und eine unstrukturierte Darstellung sorgen zudem für Unübersichtlichkeit.

■ Über die Skriptbefehle „Feld angeben“ und „AppleScript ausführen“ können Sie den Text einer Variablen als AppleScript ausführen lassen. Die Nachteile sind die gleichen wie oben, der entscheidende Vorteil aber ist: Über die Formelfunktion „Austauschen“ lassen sich Feldinhalte der Datenbank in das Skript einbetten und an andere Programme schicken.

■ Applets/Snippets sind Miniprogramme, die sich im AppleScript-Editor (im Ordner „Apple Extras“) editieren und als eigenständige Software speichern lassen. Ihr Nachteil: Der Programmstart kostet Zeit, die oft in keinem Verhältnis zum Nutzen steht. Der Vorteil liegt darin, daß ein Applet auf alle Objekte der Datenbank zugreifen kann, was immer dann sinnvoll ist, wenn mehrere Programme an der Kommunikation beteiligt sind und das Applet die Kontrolle über das Geschehen behalten soll. Dies gilt etwa, wenn Daten aus der Datenbank in ein Dokument einer anderen Applikation geschrieben und Resultate aus diesem Dokument zurückgeschrieben werden sollen.

■ Ein externes AppleScript können Sie nicht nur als ausführbare Software, sondern auch als Skript sichern. Der Nachteil: Wollen Sie auf FileMaker-Objekte Bezug nehmen, ➤

müssen Sie das Skript durch ein anderes Programm ausführen lassen, zum Beispiel durch den AppleScript-Editor.

Ein Fall für die Praxis. Wie die Zusammenarbeit zwischen dem Finder und einer FileMaker-Datei aussehen könnte, soll das folgende Beispiel einer Dokumentenverwaltung demonstrieren.

■ **Das Problem:** Für ein Projekt legen mehrere Mitarbeiter Bilddateien in einem Ordner ab. Die Bilder sollen über ihre Nummer automatisch mit bereits vorhandenen Texten und Legenden verknüpft werden. Alle Mitarbeiter hätten gern einen schnellen Zugang zum jeweiligen Status Quo.

■ **Die Aufgabe:** FileMaker soll automatisch für jedes neue Dokument einen Datensatz anlegen und das Bild über eine Referenz verknüpfen.

■ **Die Lösung:** Ein Applet vergleicht den Ordnerinhalt mit den erfaßten Datensätzen und regelt den FileMaker-Import neuer Bilder.

■ **Die Voraussetzungen:** FileMaker Pro 3.0.3, AppleScript, System 7.5, Apples Scripting-Additions sowie eine FileMaker-Datei mit den Feldern „Datei“ (Text), „Pfad“ (Text), „Bild“ (Medien) und „var_Stop“ (Variable).

■ **Die Umsetzung:** Zunächst soll die Scripting-Addition „Choose File“ nach dem gewünschten Ordner fragen. Danach beauftragt man den Finder, eine Liste aller Pict-Dateien zu liefern. Der Pfad des Ordners wird als String in die Variable „myPathS“, die Bildliste in die Variable „myFileList“ geschrieben. Wenn keine Bilddatei gefunden oder die Ordnerauswahl abgebrochen wurde, bricht die Prozedur mit einer entsprechenden Fehlermeldung ab.

Damit das folgende Skript läuft, benötigt man neben FileMaker 3.0.3 und System 7.5 auch die Osaxen „Choose File“ und „Display Dialog“:

```
set myPathA to choose folder with ~
prompt "Ordner für den Bildimport ~
auswählen ..."
tell application "Finder"
  try
    set myFileList to name of ~
    (every file of folder myPathA ~
    whose file type is "PICT")
    set mypathS to myPathA as string
  on error — Keine Pict-Datei gefunden!
```

```
beep
display dialog "Keine ~
Bilddateien gefunden!" buttons ~
"OK" default button "OK"
return
end try
end tell
```

(myFileList könnte zum Beispiel so aussehen: {"Bild 1", "Bild 2", "Bild 3", "Bild 4"}; myPathS könnte zum Beispiel so aussehen: "Macintosh HD: Bildordner:")

War der erste Schritt erfolgreich, soll FileMaker die unter diesem Pfad („myPathS“) bereits erfaßten Dateinamen liefern. Die sortierte Liste schreibt das Programm dann in die Variable „myFMList“. Unterscheidet sich die Liste von der Bildliste des Finders, erzeugen Sie über den Aufruf des FileMaker-Skripts „Alle ausschließen“ eine leere Auswahl, um die Anzeige nach dem Import auf neue Datensätze zu beschränken:

```
tell document 1 of application
"FileMaker Pro"
  try
    show (every record of database ~
    1 whose cell "Pfad" is mypathS)
    sort by field "Datei"
    set myFMList to (get cell "Datei" ~
    of every record)
  on error — Noch kein Bild mit diesem
  Pfadnamen importiert
    set myFMList to {}
  end try
  if myFMList is not myFileList then ~
  do script FileMaker script "Alle ~
  ausschließen"
end tell
```

(myFMList könnte zum Beispiel so aussehen: {"Bild 1", "Bild 2"})

Im dritten Teil vergleicht das Applet in einer Schleife die Dateiliste des Finders mit der FileMaker-Liste. Liefert diese keine Entsprechung, so wird der volle Pfadname der Datei in die Variable „myFullPathS“ geschrieben, ein neuer Datensatz erzeugt und das Bildfeld (Bild) mit der Pict-Datei verknüpft. Das Variablenfeld „var_Stop“ läßt sich in FileMaker über einen Button auf „1“ setzen, um dem Benutzer den Abbruch der Prozedur zu ermöglichen. Die letzte Zeile gibt FileMaker die Kontrolle zurück. Aus Performancegründen agiert das Applet im Vordergrund.

Wer FileMaker nach vorn holen will, muß ein „activate“ in die Zeile nach dem „tell document ...“-Statement einfügen.

```
repeat with n from 1 to count ~
(myFileList)
  if item n of myFileList is not in my ~
  FMList then
    set myFileS to item n of myFileList
    set myFullPathS to (mypathS & ~
    myFileS)
    tell document 1 of application ~
    "FileMaker Pro"
      set myRecord to create record
      go to myRecord
      set cell "Datei" of myRecord to ~
      myFileS
      set cell "Pfad" of myRecord to ~
      mypathS
    try
      set cell "Bild" of myRecord to ~
      file myFullPathS
    on error — Alias oder korrupte
    Datei
      delete myRecord
    end try
    if (get cell "var_Stop") > 0 ~
    then exit repeat — Abbruch
  end tell
end if
end repeat
tell application "FileMaker Pro" ~
to activate
```

Das Applet wird im Skripteditor als Programm unter „Bildimport“ gesichert und in FileMaker über folgendes ScriptMaker-Skript aufgerufen:

```
Wenn ["Status(AktuellPlattform) = 1"]
  Kommentar ["Client benutzt ~
  Macintosh"]
  Blättern aktivieren []
  Feld angeben ["var_Stop","0"]
  Apple Event senden
  ["Bildimport","aevt","oapp"]
Sonst
  Fehlerton
  Meldung ["Diese Funktion steht ~
  unter Windows nicht ..."]
Ende-Wenn
```

Dabei ist wichtig, darauf zu achten, daß die Option „Zielprogramm nach vorne bringen“ im Apple-Event-Editor aktiviert ist. ■ *Martin Fuchs*



Vorschau

In der nächsten Folge behandeln wir die Einbindung von AppleScript in ScriptMaker.