

Link Shop



Christian Schmitz

(Jg. 1981) entwickelt seit 11 Jahren Software und seit 2006 auch FileMaker Plugins. Außerdem Plugins für Real Studio und allerlei Software für Mac OS X, Windows, Linux und iOS.

support@monkeybreadsoftware.de

Audit per MBS-Plugin

Wie FileMaker Änderungen mitschreiben kann

Zu den zahlreichen Funktionen des MBS-Plugins gehören seit 2012 auch die Audit-Features. Da diese seit der Version 6.5 bedeutend schneller geworden sind, lohnt sich ein genauerer Blick darauf. Diese Funktionen erlauben es, alle Änderungen einer FileMaker Datenbank zu protokollieren. Dadurch sieht man nicht nur, wer was wo geändert hat, sondern kann Änderungen sogar nachträglich rückgängig machen bzw. die alten Werte anzeigen lassen.

Audit mit MBS

Um die Audit-Funktionen an einem Beispiel zu erläutern, verwende ich die FileMaker Starterlösung *Event-Management.fmp12*. Benötigt werden eine *AuditLog*-Tabelle sowie ein passendes Layout, die aber durchaus auch in einer anderen Datenbank liegen können, solange sie nur irgendwie verfügbar sind. Das MBS-Plugin ermittelt anhand des AuditLog-Layouts, welche Felder geschrieben werden sollen. Die Datensätze werden per SQL-Befehl intern gesetzt, wofür eine Beziehung zu einer Tabelle *AuditLog* bestehen muss – auch wenn diese eventuell in der Datenbank anders heißt. Das ist bei serverbasierten Lösungen durchaus interessant, wenn es mehrere *AuditLog*-Tabellen gibt und per Beziehung immer die richtige verfügbar ist.

Folgende Felder müssen in der Tabelle *AuditLog* vorhanden sein: **FieldName**, **FieldHash**, **TableName** und **RecordID**. Während der Feldname und die Tabelle das Feld eindeutig definieren, wird über die **RecordID** ermittelt, welcher

Datensatz geändert wurde und **FieldHash** ist der Hash des Feldwertes. Der Feldwert kann nämlich sehr lang sein und kann, muss aber nicht aufgeschrieben werden.

Um die verschiedensten Informationen zu erhalten, können folgende Felder definiert werden:

FieldValue, **FieldOldValue**, **FieldType**, **UserName**, **IP**, **CurrentTimestamp**, **TimeStamp**, **CurrentTime**, **CurrentDate**, **Action**, **CurrentHostTimeStamp**, **PrivilegeSetName**, **AccountName**, **LayoutNumber**, **ApplicationVersion**, **FileName**, **HostApplicationVersion**, **HostName**, **HostIPAddress**, **LayoutName**, **PageNumber**, **LayoutTableName**, **TableID**, **FieldID**, **ScriptName** und **WindowName**.

FieldValue und **FieldOldValue** werden von dem Plugin mit dem neuen bzw. dem alten Wert gefüllt. Letzterer ist natürlich nur dann verfügbar, wenn sich der ältere Eintrag zu dem Datensatz findet. In **FieldType** wird der Feldtyp hinterlegt, zum Beispiel „Text“, während **UserName** den Benutzernamen und **AccountName** den aktuellen Account-Namen enthält. In den Feldern **CurrentTimestamp**, **CurrentHostTimeStamp**, **TimeStamp**, **CurrentTime** und **CurrentDate** wird die aktuelle Uhrzeit und/oder das aktuelle Datum abgespeichert. **HostName** und **HostIPAddress** definieren, auf welchem Rechner die Änderung gemacht wurde, in **LayoutName** und **LayoutNumber** wird das verwendete Layout gespeichert und das Feld **WindowName** nimmt den Fensternamen auf. In **TableID** und **FieldID** werden die IDs für die Tabelle und das Feld hinterlegt, über die man Informationen später noch flotter finden kann.

Audit-Tabelle anlegen

Legen Sie also eine Tabelle namens *AuditLog* mit den Feldern **FieldName**, **FieldHash**, **TableName**, **RecordID**, **FieldValue**, **FieldOldValue**, **FieldType**, **UserName**, **IP**, **CurrentTimestamp**, **Action**, **FileName** und **LayoutName** sowie dem Zusatzfeld **EventID** an. In Letzterem wird die ID des Events gespeichert, sodass die Änderungen zu einem Event später gezielt zu finden sind. FileMaker legt zu der neuen Tabelle automatisch ein Layout an. Dazu benötigen Sie das aktuelle Plugin, damit die Felder in der Tabelle gefunden werden. Wir können später im Layout sehen, was gerade mitgeschrieben wurde. Für den normalen Benutzer kann die Tabelle oder das Layout natürlich unsichtbar gemacht werden.

Sie können gerne in der Tabelle *AuditLog* zusätzliche Felder anlegen, zum Beispiel **LogTime** mit dem Datentyp „Zeitstempel“, das von FileMaker automatisch gefüllt wird. Sie können diese Zusatzfelder auch via **Audit-Aufruf** später mit übergeben, zum Beispiel, um eine Variable mitzuloggen.

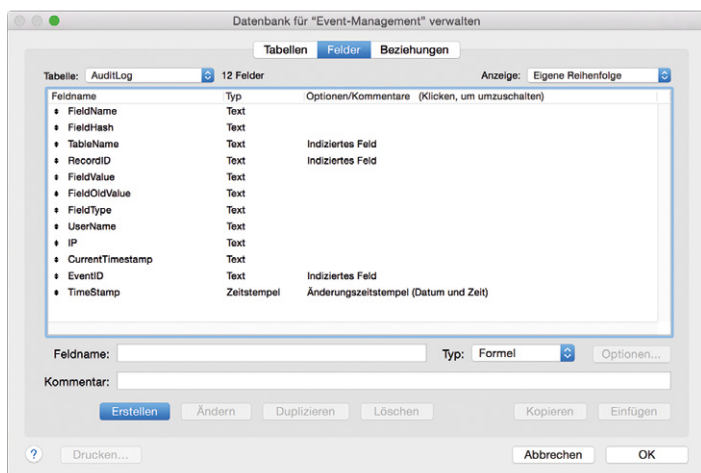


Abb. 1: Hier sehen Sie die Definition der Tabelle *AuditLog*. Die Indizes legt FileMaker später automatisch an.

Audit Felder

Damit Sie den genauen Änderungszeitpunkt nachvollziehen können, benötigen Sie in jeder Tabelle ein Feld namens **AuditTimeStamp**, in dem FileMaker bei einer Änderung am Datensatz automatisch den aktuellen Zeitstempel speichert. Falls Sie bereits ein solches Feld haben, können Sie das auch verwenden und brauchen keinen redundanten Zeitstempel.

Außerdem legen wir ein weiteres Feld namens **AuditState** an, das einen berechneten Wert enthält. Da dieser Wert immer neu gerechnet werden soll, muss die Option „Vorhandenen Feldwert nicht ersetzen“ deaktiviert sein. Die Formel zur Berechnung ist der **MBS-Aufruf** „Audit.Changed“ mit mehreren Parametern: Zuerst übergeben Sie den Zeitstempel, dahinter den Namen der Tabelle, in diesem Fall „Events“. Anschließend können Sie in den nächsten Parametern die Feldnamen angeben, die ignoriert werden sollen. Mithilfe des

Zeichens „|“, können beliebige Felder mit dem Inhalt anderer Felder gefüllt werden. Mit „EventID|“ & **EVENT_ID ABGLEICHSFELD**“ beispielsweise wird der Wert von **EventID** auf den Wert des Feldes **EVENT_ID ABGLEICHSFELD** gesetzt. Zuletzt übergeben wir den Namen des Feldes **Dauer Tage Etikett_Plural**, damit wir das Feldetikett aus der Beispieldatenbank nicht speichern. Dabei handelt es sich um ein berechnetes Feld, das leicht aus den anderen ermittelt werden kann. Der **Audit-Aufruf** sieht so aus:

```
MBS( "Audit.Changed"; AuditTimeStamp; "Events";
"EventID|" & EVENT_ID ABGLEICHSFELD; "Dauer Tage
Etikett_Plural")
```

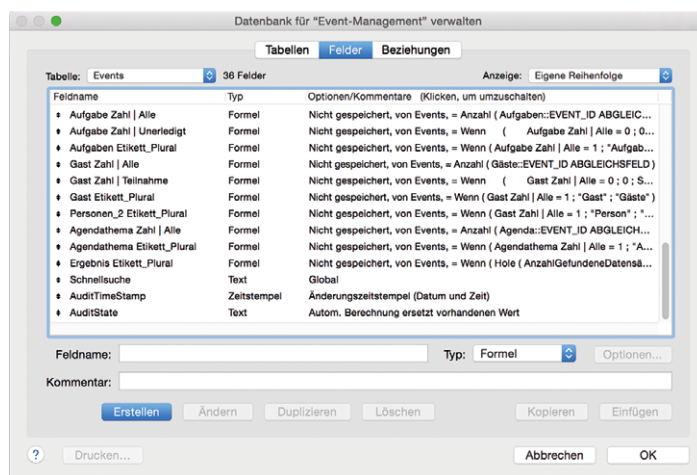


Abb. 2: In der Events-Tabelle stehen unten die zwei neuen Felder für das Audit.

Für die anderen Tabellen sehen die Aufrufe so aus:

```
MBS( "Audit.Changed"; AuditTimeStamp; "Mitwirkende";
"EventID|" & EVENT_ID ABGLEICHSFELD)
MBS( "Audit.Changed"; AuditTimeStamp; "Aufgaben";
"EventID|" & EVENT_ID ABGLEICHSFELD)
MBS( "Audit.Changed"; AuditTimeStamp; "Agenda";
"EventID|" & EVENT_ID ABGLEICHSFELD)
MBS( "Audit.Changed"; AuditTimeStamp; "Gäste";
"EventID|" & EVENT_ID ABGLEICHSFELD)
```

Man kann schön die Fenster nebeneinander offen haben, eins mit dem Events-Layout und eins mit dem Layout für die Tabelle *AuditLog*. Bei Änderungen an den Portalen muss man zum Speichern im Event auf eine Freifläche klicken, um die Änderungen am Datensatz zu schreiben. Das schreibt dann erst die Änderungen für alle Portale zusammen weg. Im *AuditLog* sehen wir viele neue Einträge für die neuen Datensätze.

Positiv oder Negativ?

Im Plugin gibt es zwei „Audit-Changed“-Befehle: „Audit.Changed“ betrifft die Feldnamen, die nicht beobachtet werden sollen, während „Audit.Changed2“ sich an einer Positiv-

liste von Feldern orientiert. Da „Audit.Changed2“ nicht erst abfragen muss, welche Felder es gibt, ist dieser Befehl der schnellere der beiden. Grundsätzlich werden Formelfelder oder ungespeicherte Berechnungen nicht überwacht, denn diese werden ja immer wieder neu berechnet.

Sie können auch bestimmte Feldtypen von der Überwachung ausschließen. Mit „Audit.SetIgnoreCalculations“ werden alle Berechnungsfelder ignoriert, „Audit.SetIgnoreSummaryFields“ lässt die Summenfelder außen vor und „Audit.SetIgnoreUnderscoreFieldNames“ exkludiert alle Felder mit Unterstrich am Wortanfang. Außerdem können Sie mit „Audit.SetIgnoredFieldNames“ eine globale Liste von Feldern definieren, die nie überwacht werden sollen. Standardmäßig ignoriert das Plugin die Felder **AuditState** und **AuditTimeStamp**.

Audit anzeigen

Wir können die Log-Einträge vom Audit direkt beim jeweiligen Datensatz anzeigen. Vielleicht nicht für jeden Benutzer, aber die Anzeige ist doch immer wieder nützlich als Änderungsprotokoll. Legen Sie dafür im Beziehungsdia-gramm eine Beziehung zwischen **Events::EVENT_ID ABGLEICHSFELD** und **AuditLog::EventID** an.

Schaffen Sie anschließend im Events-Layout etwas Platz und erstellen Sie ein Portal für die Tabelle **AuditLog**. Sollte diese nicht im Pop-up-Menü für Bezugsdatensätze zu finden sein, schauen Sie sich noch einmal die Beziehung an und korrigieren Sie diese. Für das Portal sind die Felder **FieldName**, **FieldValue** und vielleicht **UserName** ausreichend.

Im Layout sieht das jetzt so aus:

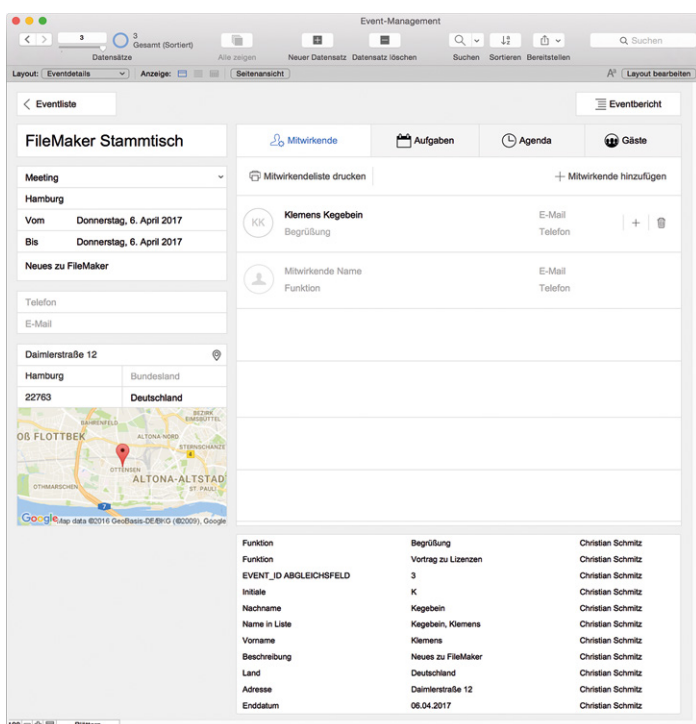


Abb. 3: Oben werden die Events und darunter die Änderungen angezeigt.

Audit beim Löschen

Um das Löschen zu protokollieren, gibt es zwei Möglichkeiten. Entweder Sie sorgen dafür, dass nur per Script gelöscht werden kann, oder Sie benutzen die Sicherheitseinstellungen in FileMaker für einen Lösch-Trigger.

Der Aufruf des Plugins erfolgt über die „Audit.Delete“-Funktion (anstelle von „MyTable“ geben Sie natürlich Ihre Tabelle an):

```
MBS( "Audit.Delete"; MyTable::AuditTimeStamp; "MyTable" )
```

Der Zeitstempel wird zwar übergeben, aber nicht wirklich verwendet. Das Plugin loggt sowohl die Änderungen (falls vorhanden) und erzeugt zusätzlich noch einen Eintrag zum Löschen. Den gleichen Aufruf können Sie jederzeit in einem Script kurz vor dem „Löschen“-Befehl selber ausführen (mit „Variable Setzen“).

Testbenutzer

Um das Ganze zu testen, legen Sie im Dialog „Sicherheit“ zunächst einen neuen Benutzer mit Lösch-Rechten an. Erstellen Sie am besten ein neues Berechtigungsset – das geht am schnellsten, wenn Sie die Berechtigungen der Dateneingabe duplizieren und anpassen. In den Einstellungen für die Berechnung gibt es links oben ein Pop-up-Menü für Datensätze. Klicken Sie dort auf „Eigene Berechtigungen ...“ und wählen Sie für alle Tabellen bei dem Punkt Löschen statt „Ja“ die Option „Eingeschränkt ...“ aus. Fügen Sie in dem folgenden Dialog zur Berechnung der Einschränkung den Pluginaufruf ein, zum Beispiel für die Events-Tabelle:

```
MBS( "Audit.Delete"; AuditTimeStamp; "Events" )
```

Zum Testen erstellen wir noch ein Script namens „Anmelden“, das nur einen Befehl beinhaltet:

```
◆ Erneut anmelden  
Mit Dialog: Ein
```

Wenn alles klappt, können Sie sich mit dem neuen Script im Handumdrehen mit dem neuen Benutzer anmelden. Legen Sie anschließend einen neuen Datensatz an und achten Sie darauf, die Änderungen zu sichern. Öffnen Sie ein zweites Fenster mit der Tabelle **AuditLog**, um die Aufzeichnung zu sehen. Wenn Sie den Datensatz löschen, wird die Berechnung ausgeführt und der Datensatz sollte als gelöscht notiert sein. Bitte beachten Sie, dass FileMaker erst die Berechnung ausführt und dann eventuell einen Dialog anzeigt. Wenn der Benutzer dort „Abbrechen“ anklickt, hat das Plugin schon einen Eintrag gesetzt und der Datensatz bleibt trotzdem ungelöscht.

Erfahrungen

Viele Pluginanwender setzen Audits ein. Da Änderungen in FileMaker im Handumdrehen abgespeichert sind, sollte jede Datenbank ein Audit haben. Schnell ist ein Datensatz geändert oder gelöscht und später weiß keiner, wer was warum gemacht hat. Außerdem kann über ein solches Audit ein Undo-Feature angeboten werden.

Die Tabelle mit dem AuditLog wird recht schnell ziemlich groß, da für jede Änderung ein Datensatz angelegt wird. Man kann die Tabelle aber leicht rotieren, also zum Beispiel monatlich wechseln. Da die Tabelle *AuditLog* und das Layout in einer offenen Datei verfügbar sein müssen (egal in welcher), empfiehlt es sich, *AuditLog* in eine andere Datei zu legen. Man kann sogar für verschiedene Dateien verschiedene Audit-Datenbanken einrichten und die Logdaten separieren, sodass mehrere Tabellen ihre jeweiligen Audit-Dateien loggen können. Dafür müssen aber die Beziehungen angepasst werden, sodass unter dem Namen „AuditLog“ ein Verweis in den Beziehungen ist, der auf die entsprechende Tabelle zeigt. Diese heißt dann natürlich anders.

Richtig interessant wird das Ganze, wenn die Benutzer nicht wissen, dass ein Audit mitgeschrieben wird. Wenn ein Mitarbeiter einen Fehler macht, kann der Vorgesetzte die Änderung zurückspielen oder bei Datenunstimmigkeiten herausfinden, wer wann an dem entsprechenden Datensatz gearbeitet hat.

Seit der Version 6.5 des MBS-Plugins liegen die SQL-Anweisungen für Audits in einer stark verbesserten Form vor, sodass das neue Plugin deutlich schneller im Netzwerk ist. Für jede Änderung in einem überwachten Feld muss natürlich im *AuditLog* nach alten Einträgen gesucht und geschaut werden, ob sich die Werte geändert haben. Probieren Sie es einfach mal aus! ■

FileMaker Magazin



Das FileMaker Magazin

- Einzige, deutschsprachige Fachzeitschrift zu FileMaker
- Wissen aus erster Hand von anerkannten FileMaker Fachautoren
- Große Themenvielfalt für Anwender und Entwickler

Exklusiv für Premium-Abonnenten

- Sechs FMM Ausgaben pro Jahr
- Kostenlose Nutzung des Abonnentenbereichs auf www.filemaker-magazin.de
- PDF-Archiv mit allen bisher veröffentlichten Ausgaben
- Jede Ausgabe mit kostenlosen Beispieldateien und Zusatzinfos zum Download

Unser Service

- Aktuelle Neuheiten, Tipps und Infos, Kleinanzeigen und vieles mehr jederzeit auf unseren Webseiten
- Hilfe bei allen Fragen zu FileMaker im FMM Forum
- Kompetente Beratung zum Kauf von FileMaker Lizenzen: Einfach anrufen +49 (0)40 589 65 79 70.

Wenn Sie sich für ein FileMaker Magazin
Abo interessieren, klicken Sie bitte hier!

Eine kostenlose **Leseprobe** des FileMaker Magazins erhalten Sie, wenn Sie hier klicken.

Hier finden Sie Aktuelles zu **FileMaker Lizenzen**, egal ob Sie kaufen, mieten oder sich einfach informieren möchten.