

***Anbindung eines RagTime-Dokumentes
an eine MySQL-Datenbank
unter Mac OS X***

von Benedikt Quirnbach
Lektoriert von Thomas Kaegi

Inhalt

Systemvoraussetzungen	3
Was ist MAMP?	3
Warum MAMP benutzen?	3
Warum nicht den in Mac OS X eingebauten WebServer Apache benutzen?	3
Was ist phpMyAdmin?.....	4
Eine Datenbank anlegen	4
MAMP starten.....	4
phpMyAdmin starten.....	5
Datenbank anlegen.....	6
Tabelle anlegen	7
Daten eintragen.....	10
Eine ODBC-Datenquelle einrichten	13
ODBC-Datenquelle in RagTime einrichten	13
ODBC-Datenquelle testen.....	15
ODBC-Datenquelle mit ODBC-Administrator einrichten.....	16
RagTime-Connect-Komponente in RagTime erzeugen	18
RagTime Connect	19
Verbindung herstellen	19
Erste Abfrage formulieren	20
Alle Daten aus der Tabelle holen.....	21
RagTime-Connect im Inventar	21
Die Adressdaten in einem Brief-Formular nutzen	22
Empfänger-Adresse anlegen.....	22
Wie gibt man solche Formeln ein?	23
Was bedeutet die Formel?	23
Was bedeutet „\$0“?	23
Durch die Datensätze „blättern“	23
Brief-Formular benutzen.....	24
Fixierung der benutzten Daten	25
Fixierung aufheben.....	25

Systemvoraussetzungen

Unter der folgenden Konstellation wurde diese Anleitung getestet:

- RagTime 6.0.2 mit RagTime-Connect
- Mac OS X 10.4.11
- MAMP 1.7.1 oder MAMP pro 1.7.1
(kostenloser Download: <http://www.mamp.info/de/index.php>)

Was ist MAMP?

MAMP ist eine Abkürzung. Sie steht für **M**acintosh, **A**pache, **M**ySQL, **P**HP. Es handelt sich dabei um eine Programmsammlung für Apple Macintosh (Mac OS X), die einen kompletten Internetserver mit Datenbank zur Verfügung stellt.

Warum MAMP benutzen?

MAMP ist kostenlos, nur für die pro-Version ist ein kleiner Betrag fällig. Für unsere Zwecke reicht die Basis-Version vollkommen aus.

Mit der Installation von MAMP bekommt man alles, was man für professionelle Datenbank-Anwendungen braucht:

- MySQL ist ein sehr mächtiger Datenbank-Server. Er ist wohl der im Internet am häufigsten genutzte Datenbank-Server überhaupt.
- Apache und PHP sind ein Gespann aus einem kompletten Webserver und einem Programm-Interpreter, der die Scriptsprache PHP verarbeitet. Auch diese Kombination dürfte im Internet am weitesten verbreitet sein.
- Durch die weite Verbreitung im Internet bekommt man über alle Komponenten sehr viele Informationen. Die Handbücher sind im Internet frei verfügbar, es gibt tausende von Tutorials, es gibt hunderte von online-Foren und Mailinglisten, die jedes erdenkliche Thema zu den Programmen behandeln.

Warum nicht den in Mac OS X eingebauten WebServer Apache benutzen?

MAMP hat den Vorteil, dass man nach dem kostenlosen Download den kompletten Server inkl. Konfiguration mit einem Klick installieren kann. Nach der Installation ist alles sofort benutzbar.

Dazu gehört auch das Datenbank-Verwaltungswerkzeug phpMyAdmin.

Was ist phpMyAdmin?

Die Programme, die mit MAMP installiert werden, besitzen keinerlei Benutzeroberfläche. Man sieht also gar nichts. Man kann sie konfigurieren, indem man jeweils eine Textdatei bearbeitet. Man kann sie bedienen, indem man ziemlich kryptische Befehle ins Terminal-Fenster tippt.

Da wir hier MAMP benutzen, erübrigen sich Konfigurationsarbeiten. Zumindest solange es sich um eine Testumgebung handelt, die nicht mit dem Internet verbunden ist. Bei Verbindungen zum Internet muss man einige Sicherheitsaspekte in der Konfiguration beachten.

Aber für die Verwaltung von Datenbanken (anlegen, löschen, Dateneintragen usw.) wäre eine durchdachte Benutzeroberfläche sehr angenehm.

Und genau das liefert phpMyAdmin. Es ist eine WebSite, die alle Aspekte der Bedienung eines MySQL-Servers abdeckt (ausgenommen die Grundkonfiguration). Um phpMyAdmin benutzen zu können, benötigt man einen WebServer und PHP. Damit wären wir wieder bei den Komponenten von MAMP.

Eine Datenbank anlegen

In dieser Anleitung gehe ich nicht auf alle Besonderheiten von Datenbanken im allgemeinen und MySQL im besonderen ein. Darüber gibt es im Internet jede Menge Informationen. Darüber hinaus wäre das ein Thema für mindestens einen Regalmeter Buch.

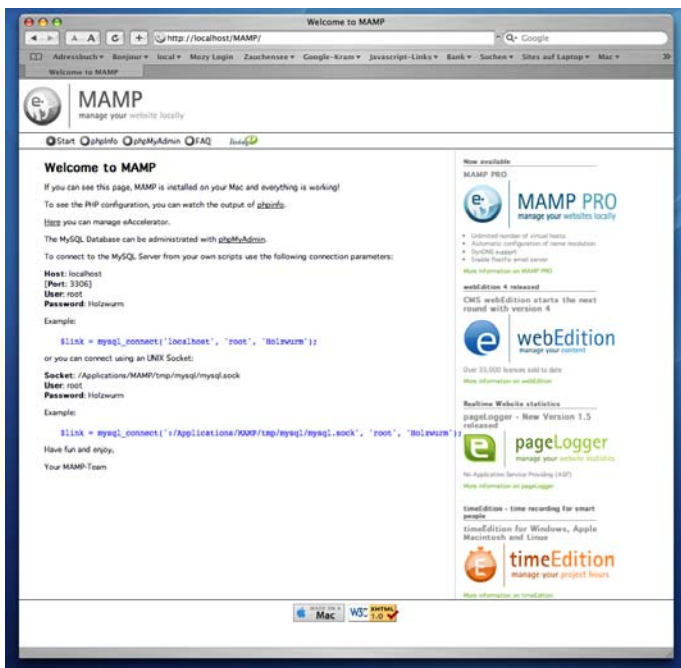
In dieser Anleitung beschränke ich mich darauf, ein kleines Beispiel zum Laufen zu bringen.

MAMP starten

Als erstes muss sichergestellt sein, dass der in Mac OS X eingebaute Webserver nicht läuft. In den Systemeinstellungen muss also unter „Sharing“ „Personal Web Sharing“ ausgeschaltet sein.

Danach kann man MAMP starten. Wie das geht, steht in der Dokumentation.

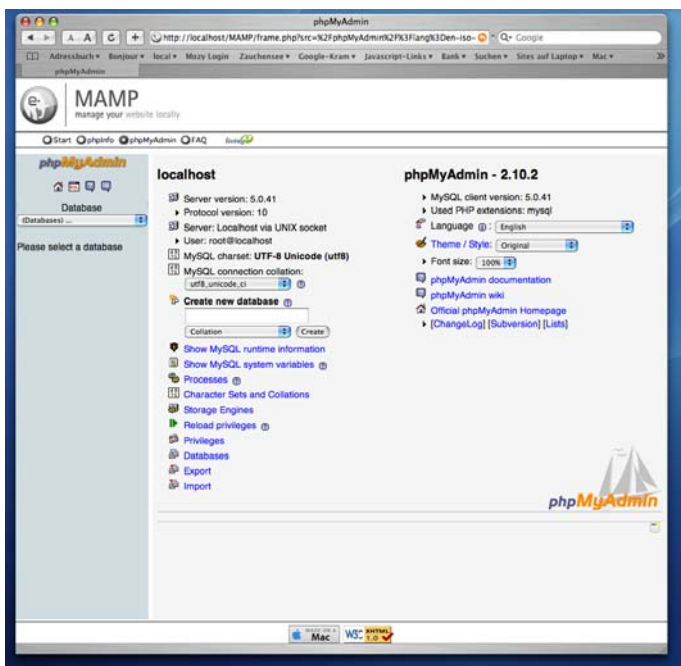
Starten Sie einen WebBrowser wie z.B. Safari. Geben Sie in die Adresszeile „localhost/MAMP“ ein.



Es erscheint die Startseite der MAMP-Installation auf Ihrem Rechner.

phpMyAdmin starten

Ein Klick auf phpMyAdmin startet die Datenbank-Verwaltung.

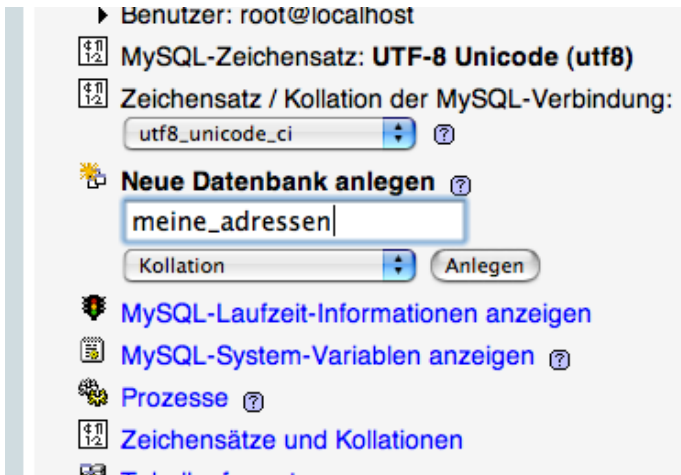


Falls die Texte nicht in deutscher Sprache erscheinen, kann man das in der rechten Spalte in einem Auswahlmeneü anpassen.

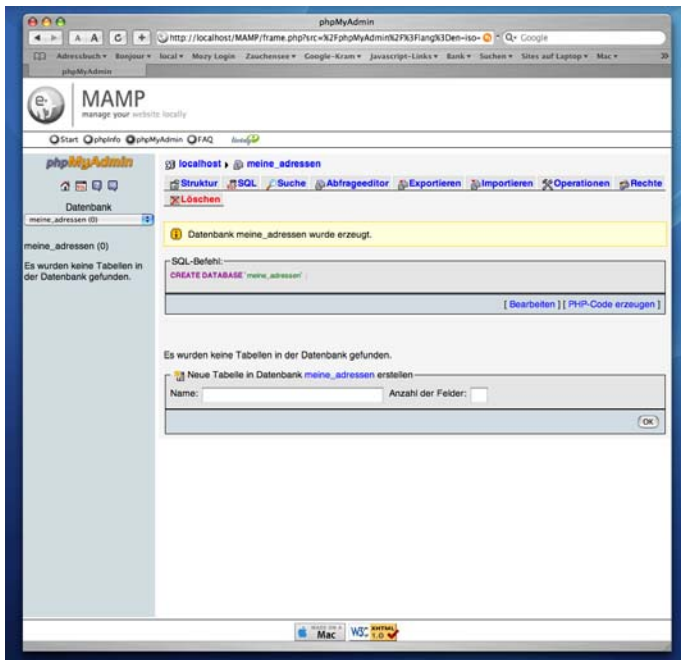
Datenbank anlegen

Für unseren ersten Test legen wir eine Datenbank mit dem Namen „meine_adressen“ an. Das wird auf der Startseite von phpMyAdmin, die Sie gerade geöffnet haben, erledigt.

Bitte beachten Sie bei allen Namensgebungen in MySQL darauf, dass keine Leerzeichen und keine Sonderzeichen (Umlaute, Satzzeichen, mathematische Operatoren usw.) enthalten sind. MySQL beachtet bei allen Namen die Schreibweise. „meine_adressen“ ist also nicht gleich „Meine_adressen“!



Geben Sie den Namen in das Textfeld unter „Neue Datenbank anlegen“ ein und klicken Sie auf „Anlegen“.

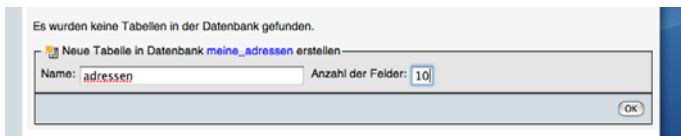


Es erscheint eine neue Seite. In der linken Spalte sehen Sie jetzt den Namen der neu angelegten Datenbank. Die Null in Klammern hinter dem Namen bedeutet, dass diese Datenbank keine Tabelle enthält.

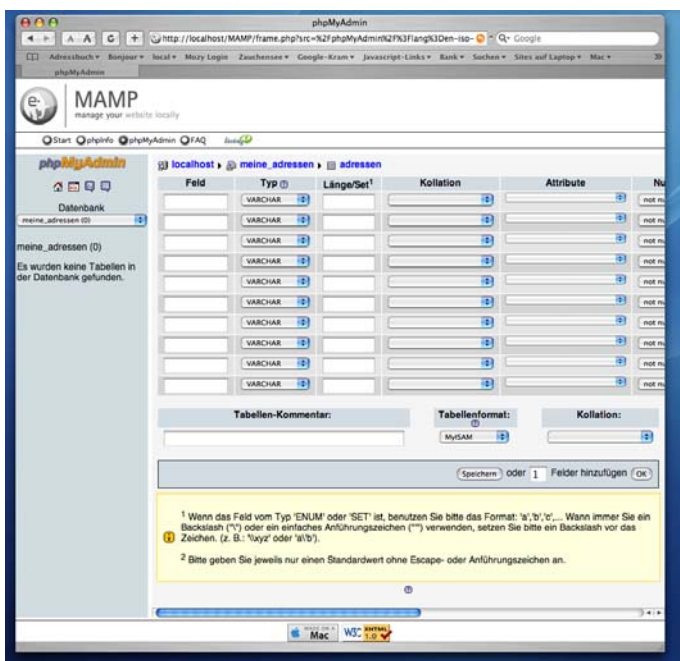
Tabelle anlegen

Jetzt haben Sie also eine Datenbank, aber noch keine Datentabelle. MySQL-Datenbanken können beliebig viele Tabellen enthalten. Für unsere Zwecke reicht eine. Sie wird den Namen „adressen“ bekommen.

Es soll eine Adress-Datenbank werden. In die einzige Tabelle können Sie später Adressdaten eingeben. Vorgesehen sind Felder für Familienname, Vorname, Strasse, Hausnummer, PLZ, Ort, Telefon, Fax und EMail. Das sind also neun Felder. Hinzu kommt noch eines, mit dem jeder Datensatz eindeutig identifiziert werden kann: id.



Geben Sie also den Tabellennamen („adressen“), die Anzahl Felder (10) ein und klicken Sie auf OK.



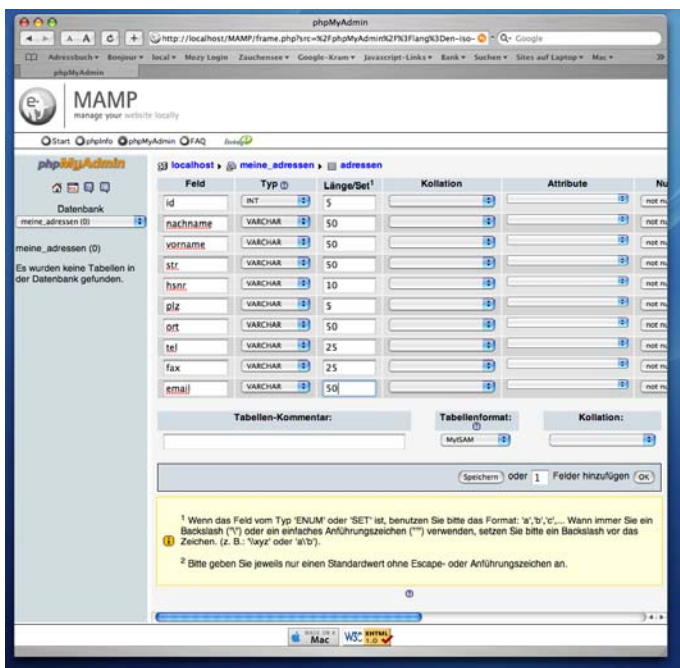
Feld	Typ	Länge/Set ¹	Kollation	Attribute	Nu
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m
	VARCHAR				not m

¹ Wenn das Feld vom Typ 'ENUM' oder 'SET' ist, benutzen Sie bitte das Format: 'a','b','c'... Wenn immer Sie ein Backslash ('\') oder ein einfaches Anführungszeichen (") verwenden, setzen Sie bitte ein Backslash vor das Zeichen. (z. B.: 'a\y' oder 'a\"b').

² Bitte geben Sie jeweils nur einen Standardwert ohne Escape- oder Anführungszeichen an.

Auf der folgenden Seite werden die einzelnen Felder der Tabelle konfiguriert. Bitte geben Sie für das Beispiel die folgenden Daten ein (Beachten Sie dabei unbedingt die Schreibweise der Feldnamen):

Feld	Typ	Länge/Set	...	Extra	„U“
id	INT	5		auto_increment	•
familiennamen	varchar	50			
vorname	varchar	50			
str	varchar	50			
hsnr	varchar	10			
plz	varchar	5			
ort	varchar	50			
tel	varchar	25			
fax	varchar	25			
email	varchar	50			



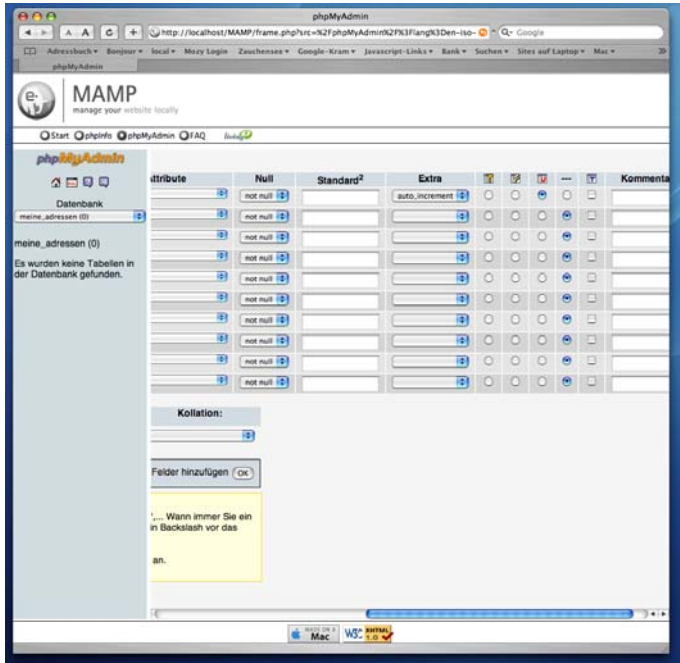
Füllen Sie das Formular wie im Bild bzw. in der Tabelle gezeigt aus. Sie sehen im Formular jeweils ein Feld für den Namen des Feldes, daneben ein Auswahlmenü für den Datentyp und danach ein Textfeld, in dem die Länge der Daten eingetragen wird.

Das Feld „id“ ist ein reines Zahlenfeld. Hier werden die Datensätze nummeriert. Für Ganzzahlen ist der Datentyp „INT“ (steht für „integer“) zuständig. Bei der eingegebenen Länge 5 sind maximal 99999 Datensätze möglich. Dabei muss man bedenken, dass die Nummer eines gelöschten Datensatzes nie wieder verwendet wird. Die maximal verwendete Nummer ist also unter Umständen sehr viel größer als die tatsächliche Anzahl der Datensätze.

Die anderen Felder sind alle Textfelder. Dafür gibt es verschiedene Datentypen, die je nach erwarteter Textmenge verwendet werden. Der Datentyp „varchar“ (für „various characters“) ist für kleine Textmengen gedacht. Im Feld „Länge“ gibt man jeweils an, wie lang der eingegebene Text maximal sein darf (Anzahl Zeichen). Werden längere Texte eingegeben, werden sie an der angegebenen Maximallänge kommentarlos abgeschnitten. Man kann maximal 256 Zeichen in ein solches Feld eintragen. Die von mir

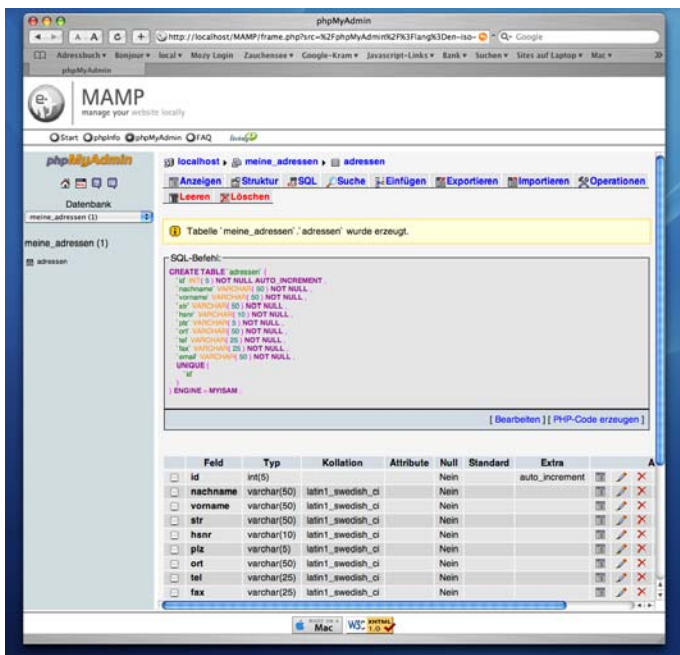
vorgeschlagenen Längen bei den einzelnen Feldern sind relativ willkürlich gewählt. Mit diesen Angaben habe ich bisher gute Erfahrungen gemacht.

Damit das Feld „id“ auch wie gewünscht funktioniert (Datensätze so nummerieren, dass jede Nummer nur einmal vorkommt), müssen hier noch zwei Einstellungen vorgenommen werden.

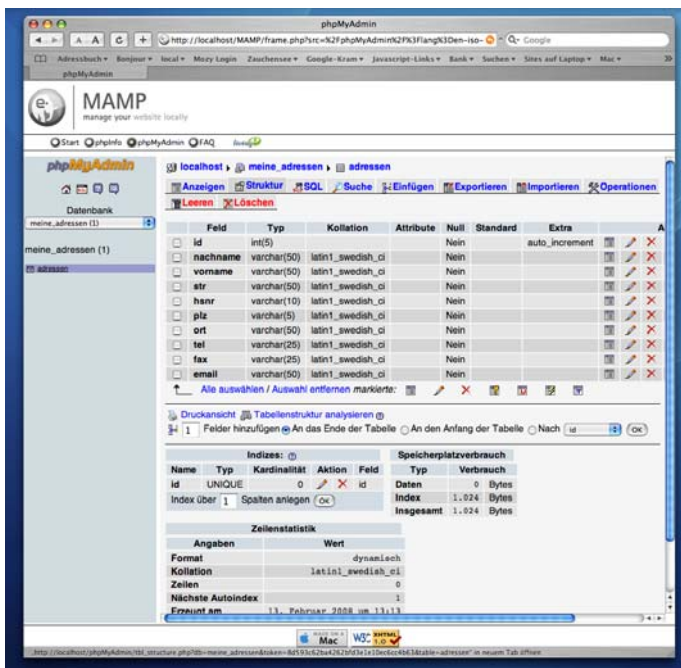


Wählen Sie für dieses Feld (und kein anderes) unter „Extras“ „auto_increment“ (automatisches hinaufzählen) und den Radiobutton unter dem Ikon mit dem „U“ (unique - einzigartig) aus. Die beiden Einstellungen befinden sich ziemlich weit rechts im Formular, also ggf. dorthin scrollen.

Klicken Sie auf „Speichern“.



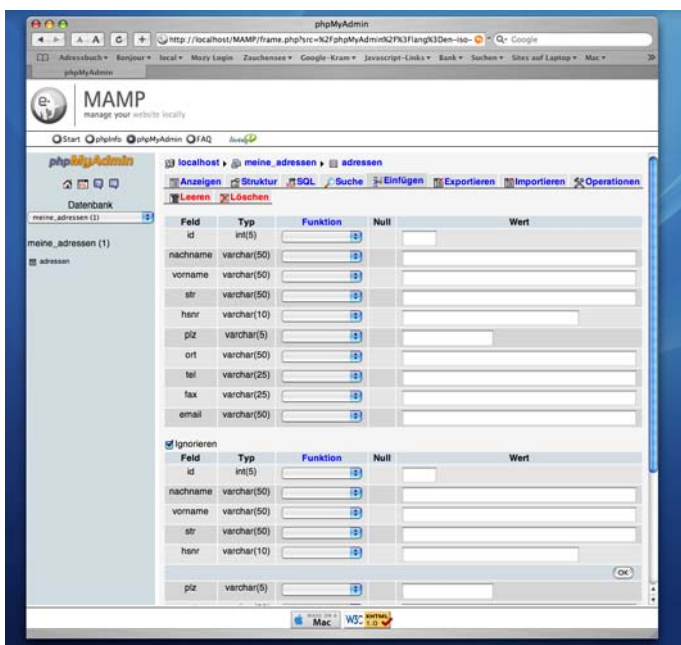
Es erscheint eine neue Seite.



Sie können jetzt in der linken Spalte von phpMyAdmin unter dem Namen der Datenbank (jetzt mit einer 1 in Klammern) den Namen der neuen Tabelle sehen. Klicken Sie darauf. Sie sehen die oben abgebildete Seite.

Daten eintragen

Zum Eintragen von Daten klicken Sie auf den Reiter „Einfügen“. Es erscheint die folgende Seite:



Tragen Sie hier die Daten ein. Achten Sie darauf, dass das **Feld „id“ immer leer** bleibt. Die Nummerierung der Datensätze nimmt MySQL automatisch vor. Da sollten Sie auf keinen Fall eingreifen.

Für das Beispiel habe ich folgende Daten eingegeben. Sie müssen nicht die selben Daten verwenden. Für das Beispiel wäre es aber gut, wenn Sie mehrere Personen mit dem selben Familiennamen eintragen und auch hier und da die Felder email, fax oder tel leer lassen würden.

familiennname	vorname	str	hsnr	plz	ort	tel	fax	email
Quirnbach	Benedikt	Musterstr.	13	32569	Musterstadt	02365 147 89 98	02365 147 89 99	bq@muster.de
Kaegi	Thomas	Musterstr.	45	32569	Musterstadt	02365 698 254 7	02365 698 254 8	tk@muster.ch
Müller	Luise	Luisenstr.	157 - 159	02365	Ostheim	0369 85 47	0369 58 71	l.mueller@muellerostheim.de
Müller	Karl-Heinz	Luisenstr.	157 - 159	02365	Ostheim	0369 85 47	0369 58 71	kh.mueller@muellerostheim.de
Schmitz	Fritz	Am Anger	5	98745	Dahinten	02587-147 85 23		
Menge	Marga	Feldweg	3	45896	Aufmdorf	03497-852 97 13		

Eine ODBC-Datenquelle einrichten

Damit RagTime auf die Datenbank zugreifen kann, müssen zwei Vorkehrungen getroffen werden:

- Im System muss eine ODBC-Datenquelle registriert sein und
- im RagTime-Dokument muss eine RagTime-Connect-Komponente erzeugt werden.

Man kann unter Mac OS X zwei Wege nehmen:

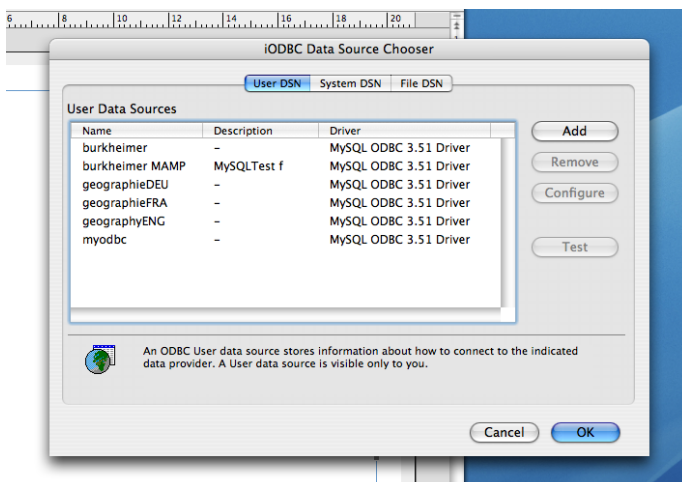
- Beide Schritte in RagTime durchführen,
- Datenquelle in ODBC-Administrator einrichten, RagTime-Connect-Komponente in RagTime einrichten

Ich würde den Weg über RagTime nehmen, da es viel komfortabler ist. Es ist allerdings bei verschiedenen Versuchen zu Problemen gekommen. Sollte es also über RagTime nicht möglich sein, eine Datenquelle einzurichten, probieren Sie den anderen Weg.

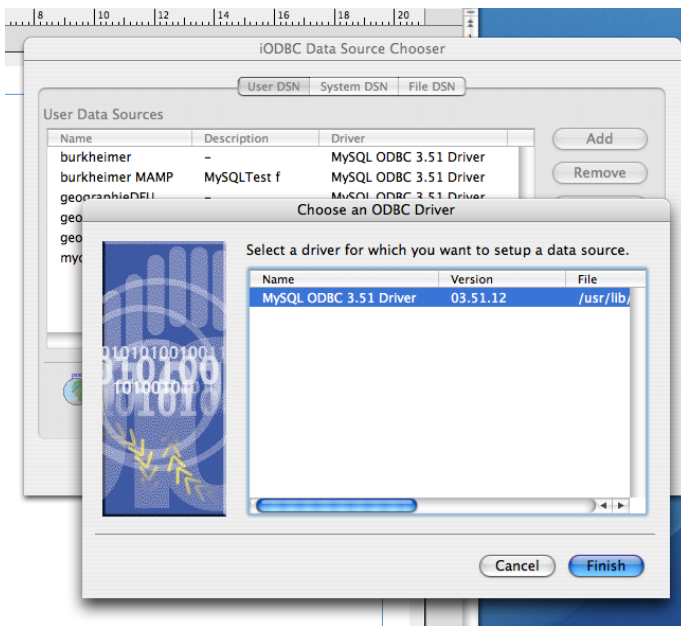
ODBC-Datenquelle in RagTime einrichten

Als erstes öffnen Sie ein neues RagTime Dokument und erzeugen eine neue RagTime-Connect-Komponente (Fenster -> Neue Komponente -> RagTime Connect).

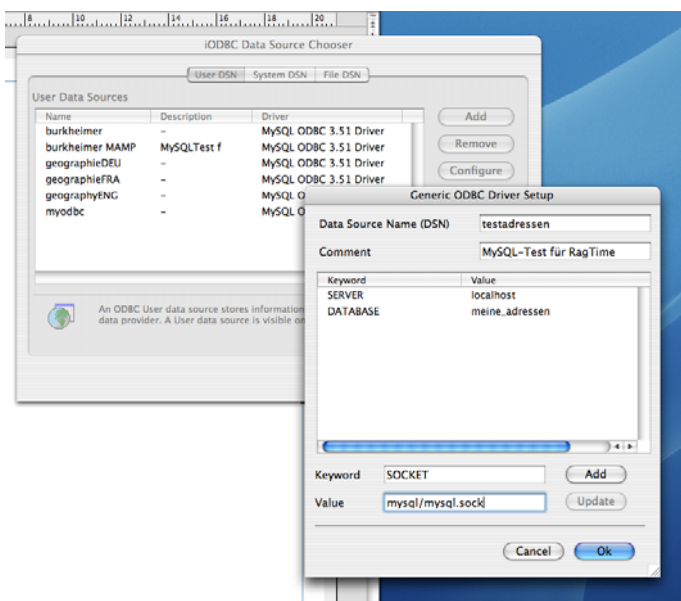
Es erscheint ein Fenster zur Auswahl oder Einrichtung einer ODBC-Datenquelle. Wählen Sie die Rubrik „Benutzer-DSN“.



Klicken Sie auf „Add“.



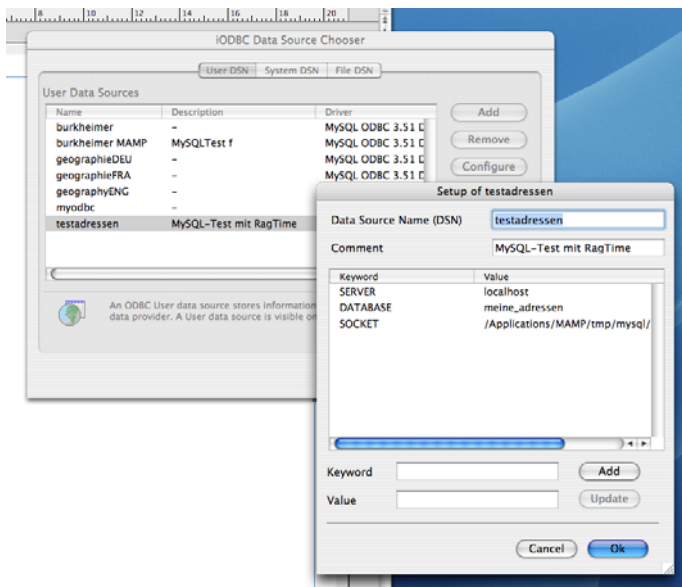
In einem neuen Fenster wählen Sie den MySQL-Treiber aus (ist bei mir der einzige vorhandene Treiber). Klicken Sie auf „Finish“.



In dem folgenden Fenster richten Sie die Datenquelle ein: Geben Sie ihr den Namen „testadressen“ und evtl. einen Kommentar, damit sie später noch wissen, wozu sie gut ist.

Dann müssen Sie in den unteren Textfeldern drei Keyword-Wert-Paare eingeben. Geben Sie dafür die Paare jeweils ein und klicken Sie dann auf „Add“. Die Paare lauten:

- SERVER localhost
- DATABASE meine_adressen
- SOCKET /Applications/MAMP/tmp/mysql/mysql.sock



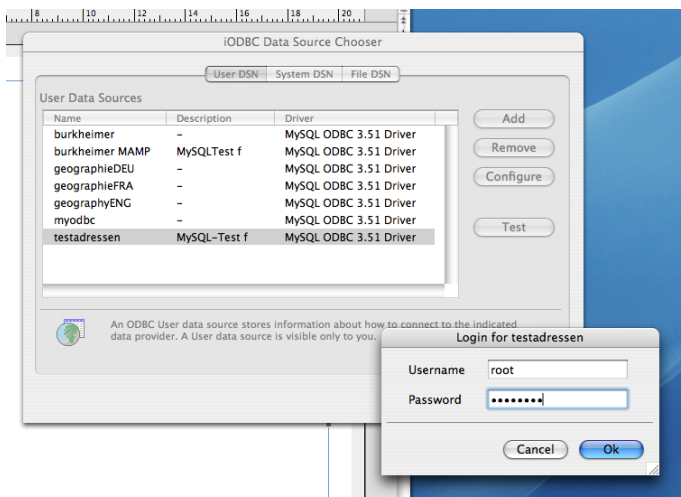
Diese Parameter geben an, dass die Datenbank auf dem eigenen Rechner liegt (localhost), wie die Datenbank heißt (meine_adressen) und wo die Socket-Datei zu finden ist.

Klicken Sie auf „OK“.

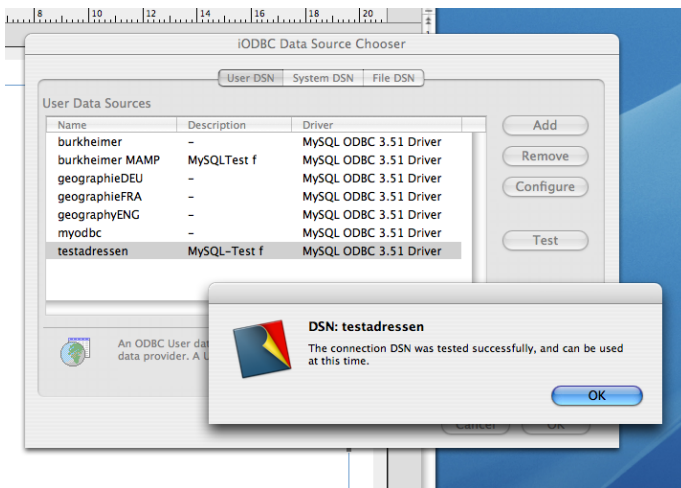
Die Datenquelle ist damit eingerichtet.

ODBC-Datenquelle testen

Um sicher zu gehen, dass auch alles funktioniert, sollten Sie die Verbindung testen. Wählen Sie dafür in der Liste der Datenquellen die neue Datenquelle aus (einmal draufklicken) und klicken Sie danach auf „Test“.



Sie werden aufgefordert Ihre Login-Daten für den MySQL-Server einzugeben. Wenn Sie die Konfiguration von MAMP nicht geändert haben, sollte das „root“ und „root“ sein. Klicken Sie auf „OK“.



Wenn alles gut gegangen ist, sollte eine entsprechende Meldung erscheinen.

Klicken Sie in der Meldung auf „OK“ und anschließend im Datenquellen-Fenster auf „OK“.

ODBC-Datenquelle mit ODBC-Administrator einrichten

Sollten Sie beim Test der Verbindung nicht erfolgreich sein, obwohl Sie alles wie beschrieben eingetragen haben, versuchen Sie diesen Weg.

In den Dienstprogrammen das Programm ODBC-Administrator starten und die Rubrik „Benutzer-DSN“ wählen.



Links unten auf das Schloss klicken, um Änderungen zu ermöglichen und mit dem Administrator-Passwort bestätigen, dass man dazu berechtigt ist.

Auf „Hinzufügen“ klicken, im Unterfenster den MySQL-Treiber auswählen und auf OK klicken:



Im nächsten Unterfenster zuerst der Datenquelle den Namen „testadressen“ geben und mit einer kurzen Beschreibung den Zweck charakterisieren. Danach auf „Hinzufügen“ klicken. In der Liste erscheint eine Anzeige mit den Worten „Schlüssel“ und „Wert“.



Das Wort „Schlüssel“ doppelklicken und „SERVER“ eingeben. Mit Tabulator zum „Wert“ wechseln und „localhost“ eingeben. **Achtung: Jetzt darf nicht auf OK geklickt werden!** Mit „Hinzufügen“ muss eine zweite Zeile in die Liste aufgenommen werden mit „DATABASE“ und dem Namen der Datenbank, also in unserem Beispiel „meine_adressen“. Erneut nicht auf OK, sondern „Hinzufügen“ klicken und als dritten Listeneintrag „SOCKET“ und den Pfad zur Socketdatei eingeben, in unserem Fall: „/Applications/MAMP/tmp/mysql/mysql.sock“



Die Paare lauten:

- SERVER localhost
- DATABASE meine_adressen
- SOCKET /Applications/MAMP/tmp/mysql/mysql.sock

Danach unbedingt unterhalb der Liste in den leeren Raum klicken um den dritten Eintrag sicher abzuschliessen. Mit den drei so fertig eingegebenen Listeneinträgen kann die Eingabe jetzt mit OK abgeschlossen werden. Damit wird im Hauptfenster die Datenquelle eingetragen.

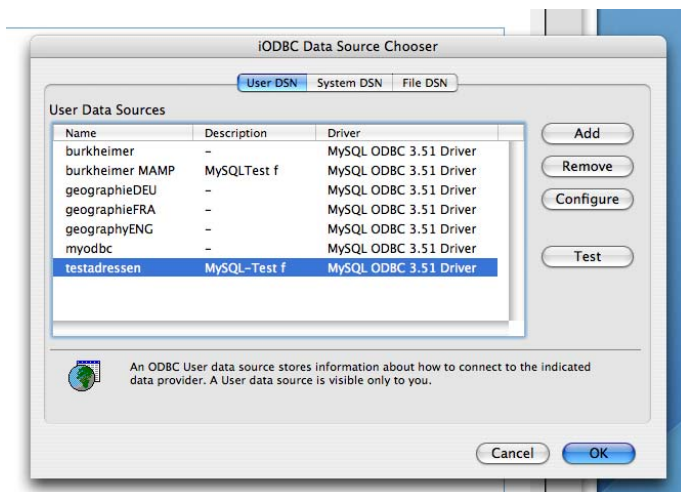


Wirklich übernommen wird sie aber erst mit Klick auf „Anwenden“. Dieser Klick ist zwingend! Sonst kann der neue Eintrag ohne Vorwarnung beim Klick auf das Schloss gleich wieder verschwinden und die ganze Mühe war umsonst! Zum Schluss das Schloss wieder verriegeln und das Fenster schließen.

RagTime-Connect-Komponente in RagTime erzeugen

Jetzt öffnen wir ein neues RagTime Dokument und erzeugen eine neue RagTime-Connect-Komponente (Fenster -> Neue Komponente -> RagTime Connect).

Es erscheint ein Fenster zur Auswahl oder Einrichtung einer ODBC-Datenquelle. Wählen Sie die Rubrik „Benutzer-DSN“.

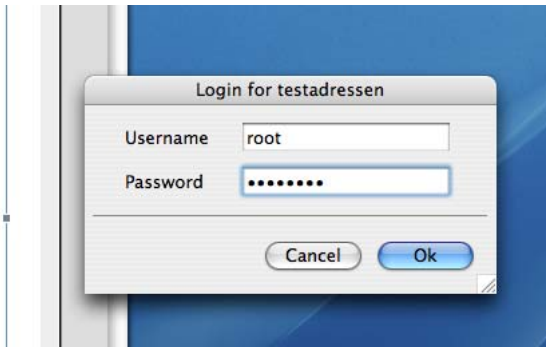


Wählen Sie die eben angelegte Datenquelle aus, indem Sie einmal darauf klicken. Klicken Sie auf „OK“.

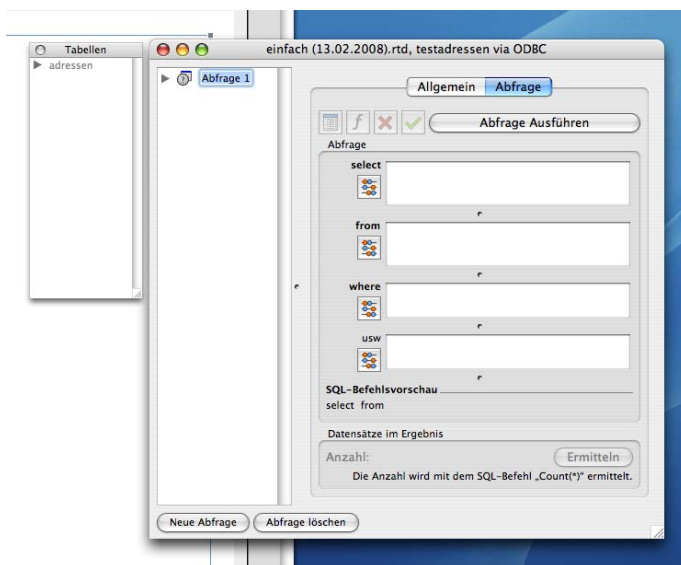
RagTime Connect

Verbindung herstellen

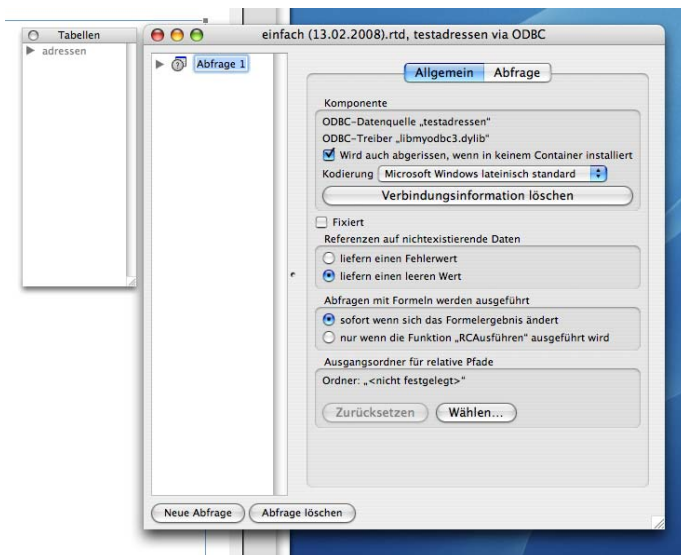
Sie müssen sich noch einmal in den MySQL-Server einloggen, damit RagTime die Komponente öffnen kann.



Nach erfolgreichem Login steht die RagTime-Komponente zur Verfügung und kann konfiguriert werden.



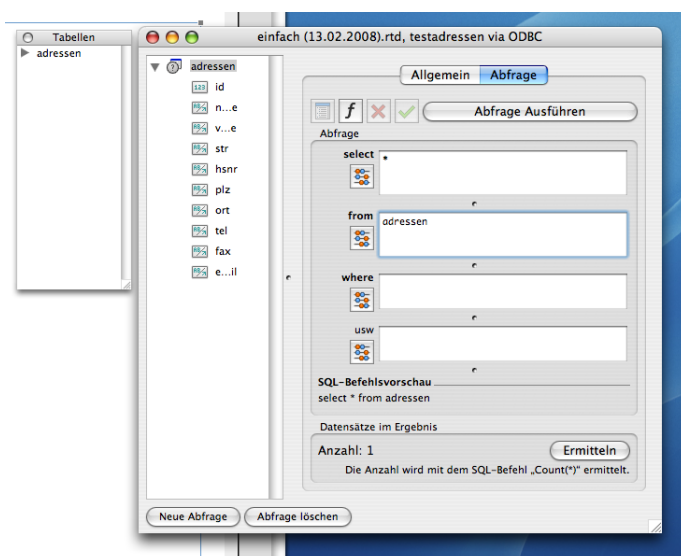
Zunächst sollten zwei Einstellungen in der Tafel „Allgemein“ vorgenommen werden.



Da MySQL ein plattform-übergreifender Datenbank-Server ist, muss er mit vielen verschiedenen Betriebssystemen auskommen. Dabei gibt es immer wieder Probleme mit der Zeichenkodierung. So ist das auch bei mir. Ich muss die Zeichenkodierung in der Tafel „Allgemein“ auf „Microsoft Windows lateinisch standard“ einstellen. Dann kommen auch die Sonderzeichen aus der Datenbank korrekt an.

Die zweite Einstellung betrifft die Fälle, wo Datenfelder leer sind. Man bekommt dann entweder einen Fehlerwert oder nichts. Ich stelle hier „liefert leeren Wert“ (also nichts) ein.

Erste Abfrage formulieren



MySQL wird mit der Sprache SQL (structured query language) betrieben. Ich kann hier keine Einführung in SQL geben. Darüber gibt es viele Bücher und auch Tutorials im Internet.

Alle Daten aus der Tabelle holen

Als erste einfache Abfrage holen wir alle Daten aus der Datenbank. Dafür wechseln wir auf die Tafel „Abfrage“. Hier wird mit ein paar Eingabefeldern die SQL-Syntax schon vorgegeben.

Die Abfrage lautet in SQL: `SELECT * FROM adressen`

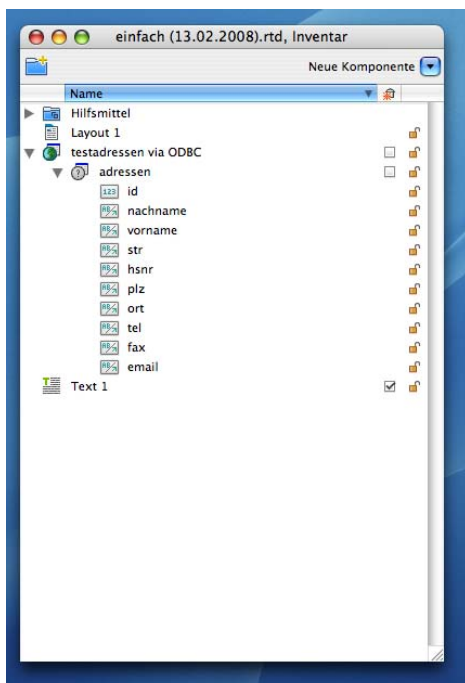
Das heißt: Wähle alles (*) aus der Tabelle adressen

In RagTime Connect gibt man dem entsprechend in das Feld „SELECT“ ein Sternchen ein. Dann klickt man in das Feld „FROM“ und dann in der Palette „Tabellen“ auf den Namen der Tabelle. (Man könnte den Namen der Tabelle natürlich auch selbst eintippen...)

Dann klickt man auf „Abfrage ausführen“ und in der linken Spalte erscheinen alle Datenfelder der Tabelle. Ein Klick auf „Ermitteln“ zeigt an, wie viele Datensätze durch diese Abfrage geliefert werden.

RagTime-Connect im Inventar

Danach kann man das Fenster der Komponente schließen. Im Inventar sieht das dann so aus:



Wie mit der Komponente umzugehen ist steht im Handbuch von RagTime Connect. Im Folgenden kommt aber noch ein kurzes Beispiel.

Die Adressdaten in einem Brief-Formular nutzen

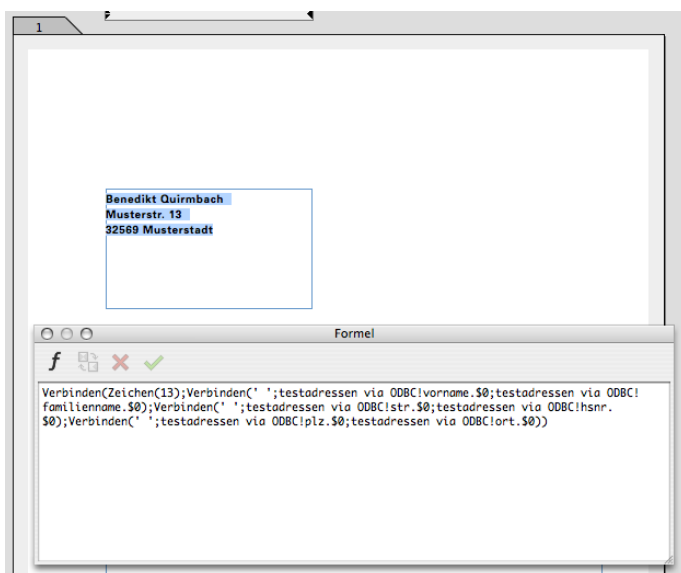
Wenn Sie bis hierher der Anleitung gefolgt sind, haben Sie jetzt ein RagTime-Dokument mit einer RagTime-Connect-Komponente.

Ich möchte in dieser Anleitung keine Einführung in die Benutzung von RagTime-Connect geben. Das wäre sicherlich eine weitere Anleitung wert. Zum Abschluss möchte ich hier kurz beschreiben, wie man am einfachsten die gerade angelegte Adressdatenbank benutzen kann.

Empfänger-Adresse anlegen

Ziehen Sie im Layout des Dokumentes einen Container auf und stellen seine Inhaltsart auf Text. Klicken Sie hinein und rufen Sie die Formelpalette auf. Klicken Sie in die Formelpalette und geben Sie die folgende Formel ein (aber in einer Zeile):

```
Verbinden(  
Zeichen(13);  
testadressen via ODBC!vorname.$0 && testadressen via ODBC!familienname.$0);  
testadressen via ODBC!str.$0 && testadressen via ODBC!hsnr.$0;  
testadressen via ODBC!plz.$0 && testadressen via ODBC!ort.$0  
)
```



Wie gibt man solche Formeln ein?

Die Formel sieht recht kompliziert aus, ist aber relativ leicht einzugeben:

Tippen Sie alle Texte mit der Hand ein bis Sie an die Stellen kommen, wo auf die RagTime-Connect-Komponente Bezug genommen wird. Klicken Sie dann im Inventar auf den Feldnamen in der RagTime-Connect-Komponente.

Fangen Sie also mit „Verbinden(Zeichen(13);Verbinden(' ');“ an. Klicken Sie auf den Feldnamen „vornamen“. Dann setzt RagTime automatisch „testadressen via ODBC!vorname.\$0“ ein. Das funktioniert also ähnlich wie Referenzen in Rechnblättern.

Sie können auch die Formel erst in einer Text-Komponente eingeben, dann kopieren und in der Formel-Palette einsetzen. Das ist unter Umständen ganz praktisch, hat aber den Nachteil, dass Sie die Referenzen auf die Datenfelder selbst schreiben müssen.

Was bedeutet die Formel?

Die Funktion „Verbinden()“ benötigt mindestens drei Parameter (die Angaben in der Klammer). Der erste Parameter gibt einen Text an, mit dem die folgenden, durch Semikolon getrennten Parameter verbunden werden sollen.

Aus „Verbinden(' - ';'Rot';'Grün')“ wird „Rot - Grün“. RagTime kann so beliebig viele Textstücke mit einem Verbindungstext (erster Parameter) verbinden.

Eine Adresse besteht aus mehreren Zeilen. Zeilenumbrüche werden in Formeln mit der Funktion „Zeichen()“ und dem Parameter „13“ - also „Zeichen(13)“ - erzeugt.

Die einzelnen Zeilen der Adresse bestehen wiederum aus zusammen gesetzten Texten. Das geht am einfachsten durch doppelte „&“: „'Anton' && 'Müller')“ ergibt „Anton Müller“. Das „&&“ erzeugt zwischen den beiden Texten ein Leerzeichen. Sollte einer der beiden Texte leer sein, wird kein Leerzeichen erzeugt.

Was bedeutet „\$0“?

Die einzelnen Datenfelder werden - wenn Sie so vorgehen, wie beschrieben - mit „\$0“ referiert: „testadressen via ODBC!vorname.\$0“.

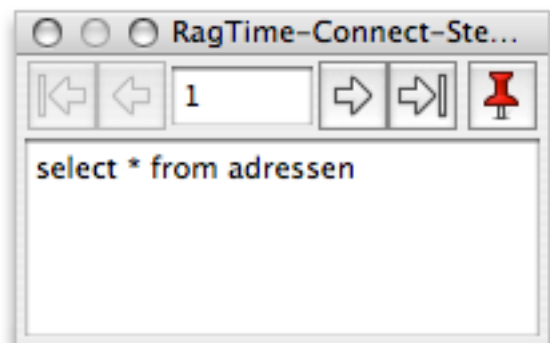
Am Ende der Referenz steht immer die Nummer des Datensatzes (nicht zu verwechseln mit der „id“ im Datensatz!). „\$0“ ist in RagTime-Connect eine besondere Referenz. Es handelt sich dabei um den Datensatz des Abfrage-Ergebnisses, der gerade ausgewählt ist. Man kann durch das Abfrage-Ergebnis „blättern“, also einen Datensatz nach dem anderen aufrufen, sofern man mehr als einen Datensatz geliefert bekommt. „\$0“ ist immer eine Referenz auf den Datensatz, den man gerade „aufgeblättert“ hat.

Durch die Datensätze „blättern“

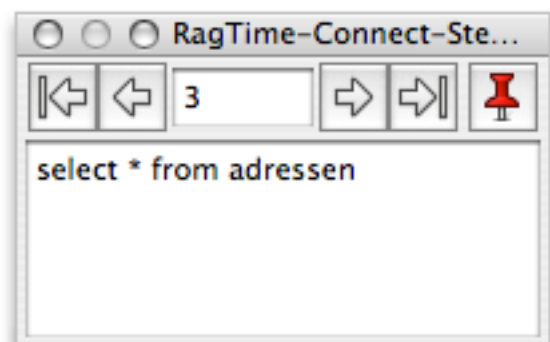
Bei unserer Abfrage werden alle Datensätze aus der Adresstabelle geholt. Man hat also ein Ergebnis mit mehreren Datensätzen. Im Adressfeld soll aber natürlich immer nur eine Adresse angezeigt werden. Das ist zunächst durch die \$0-Referenz in der Formel die erste Adresse im Abfrage-Ergebnis.

Mit Hilfe der Palette RagTime-Connect-Steuerung (Fenster -> Paletten - RagTime-Connect-Steuerung) kann man durch das Abfrage-Ergebnis bättern bis zu der Adresse, die man für seinen Brief benötigt.

Benedikt Quirnbach
Musterstr. 13
32569 Musterstadt



Karl-Heinz Müller
Luisenstr. 157 - 159
02365 Ostheim



Speichern Sie das Brief-Formular als RagTime-Formular.

Brief-Formular benutzen

Öffnen Sie das gerade gespeicherte Formular. Öffnen Sie die RagTime-Connect-Steuerung (Fenster -> Paletten - RagTime-Connect-Steuerung).

Blättern Sie zu einer neuen Adresse.

Sichern Sie das Dokument.

Fixierung der benutzten Daten

Beachten Sie, dass die Auswahl des aktuellen Datensatzes im Dokument nicht fixiert ist. Das ist eine Besonderheit von RagTime-Formularen. In „normalen“ Dokumenten wird die aktuelle Datenbank-Auswahl beim Schließen fixiert. Das heißt: Beim nächsten Öffnen befinden sich genau die gleichen Texte im Dokument wie beim Schließen. Das trifft auch zu, wenn sich die entsprechenden Datensätze in der Datenbank zwischenzeitlich geändert haben sollten oder die Datenbank nicht erreichbar ist (ist gelöscht oder der Server ist nicht erreichbar).

Den Sinn dieses Verhalten kann man am einfachsten in einem Beispiel erklären:

Stellen Sie sich vor, Sie haben eine Rechnung geschrieben, die die Preise der einzelnen Posten aus einer Datenbank holt. Sie speichern die Rechnung zur Archivierung. Wenn Sie später die Rechnung ansehen, müssen darin genau die selben Preise stehen wie bei Rechnungsstellung. Die Preisangaben in der Datenbank könnten sich aber zwischenzeitlich geändert haben. Das selbe trifft natürlich auf alle möglichen Daten zu wie z.B. auch Adressdaten.

Außer durch das Sichern können Sie die Daten auch bei noch geöffnetem Dokument z.B. in der Palette RagTime-Connect-Steuerung durch Klick auf das rote Nadel-Symbol fixieren.

Fixierung aufheben

Wenn Sie dieses Dokument wieder öffnen, ist die Auswahl des Datensatzes fixiert. Sie können also ohne weitere Vorkehrungen keinen anderen Datensatz auswählen.

Wenn Sie hier einen neuen Datensatz auswählen möchten, müssen Sie erst die Fixierung aufheben. Das können Sie entweder in der RagTime-Connect-Steuerung oder indem Sie die Komponente im Inventar doppelklicken - also öffnen - und dann in der Tafel „Allgemein“ das Häkchen bei „ist fixiert“ wegeklicken.