

AP-Änderungsprotokoll-Manager Pro

(Modul zum Protokollieren und Historisieren von Datensatzänderungen und Löschungen)

Den AP-Änderungsprotokoll-Manager in eigene Access-Anwendungen eingebaut, protokolliert das Modul alle Neuanlagen, Änderungen und Löschungen an den Datensätzen. Das Modul ist ereignisgesteuert und kann dadurch direkt in die Anwendungsformulare eingefügt werden.

In dieser Anleitung erklären wir Schritt für Schritt, wie das Modul in eine eigene Anwendung eingebaut wird. Der Einbau ist sehr einfach. Es müssen nur einige Tabellen, Formulare und Module importiert und ein paar Codezeilen in vorhandene Formulare eingefügt werden. Jedes Formular, in welchem die Protokollierung aktiv sein soll, benötigt einige Funktionen, die als ganzer Block in den VBA-Bereich des Formulars kopiert werden können. Wichtig für die Protokollierung ist, dass es bei der zugrunde liegenden Datenherkunft (Tabelle oder Abfrage) einen eindeutigen Schlüssel (Autowert, Primary Key) gibt. Dieser wird dafür verwendet, dass die protokollierten Daten dem Datensatz wieder zugeordnet werden können. Beim AP-Änderungsprotokoll-Manager **PRO** können auch Daten in Unterformularen der ersten Ebene protokolliert werden. Wird ein Datensatz des Hauptformulars gelöscht, werden dabei auch alle Daten der angehängten Datensätze der Unterformulare historisiert. Bei diesen Unterdatensätzen ist ebenfalls ein eindeutiger Schlüssel (Autowert, Primary Key) notwendig.

Für den Einbau des AP-Änderungsprotokoll-Managers sind vorhandene Kenntnisse in der Access-VBA-Programmierung von Vorteil, jedoch nicht zwingend erforderlich. Lesen Sie sich diese Anleitung bitte genau durch und folgen Sie den Anweisungen bitte auch im Detail. Selbst für einen Access-Anwender oder Access-Power-User dürfte der Einbau zu schaffen sein. Sollten Sie keine Erfahrung mit Access haben und auch mit unserer Anleitung nicht zurecht kommen, bieten wir einen Integrationservice an. Bitte kontaktieren Sie uns in diesem Fall per Mail und wir erstellen Ihnen nach Absprache der Details ein attraktives Angebot.

Infos zum Integrationservice:

Mail bitte an: webmaster@access-paradies.de

Betreff: Integrationservice für den AP-Änderungsprotokoll-Manager

Mailinhalt:

Beschreiben Sie bitte, in welche Art von Anwendung das Modul eingebaut werden soll, welche Access-Version Sie verwenden und wie groß die Datenbank im gezippten Zustand ist. Teilen Sie uns bitte mit, welche Möglichkeiten Sie haben, um uns die Datenbank zukommen zu lassen (Mail, Download vom Internet, CD usw.)

Sie können uns Ihre Anfrage auch faxen oder per Post zusenden.

Fax: (08 21) 2 99 10 72

Microsys Kramer
Günther Kramer
Ludwig-Thoma-Str. 7
D-86368 Gersthofen

Verweise:

Bevor Sie das Modul in Ihre Anwendung integrieren können, müssen bestimmte Verweise verfügbar sein. Prüfen Sie deshalb zunächst die verfügbaren Referenzen in Ihrer Anwendung.

Wie prüft man die vorhandenen Verweise?

Öffnen Sie Ihre Anwendung. Im Datenbankfenster klicken Sie auf den Karteireiter **Module**. Öffnen Sie im Entwurf ein vorhandenes Modul oder erstellen Sie ein neues Modul. Über das Menü **Extras / Verweise** öffnen Sie den Verweisdialog.

Folgende Verweise müssen aktiviert sein:

Für Access 97:

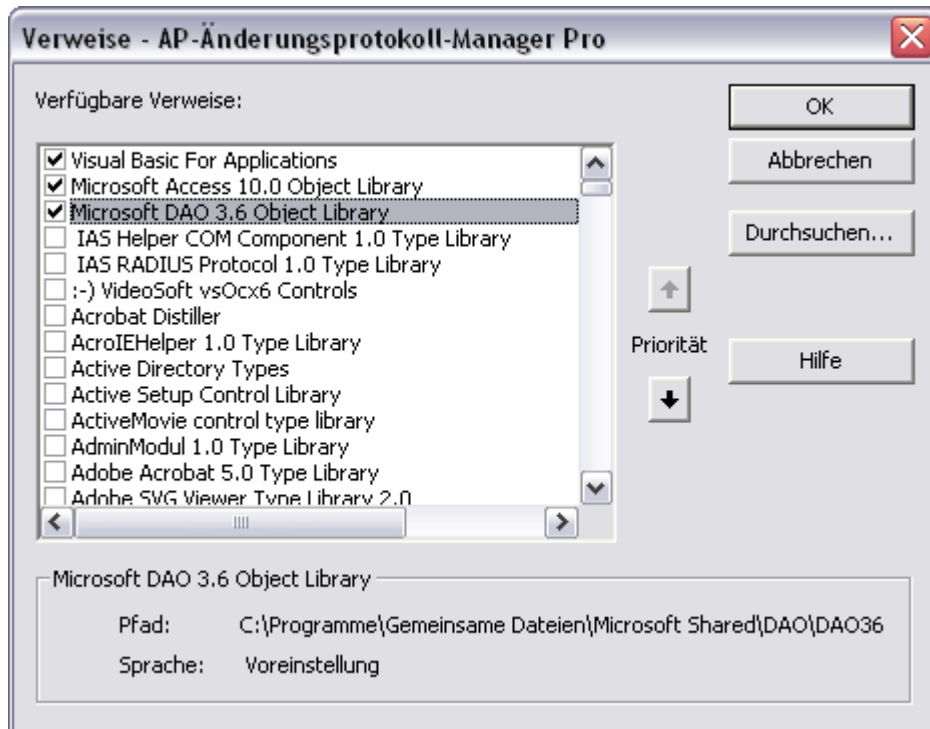
- Visual Basic For Applications
- Microsoft Access 8.0 Object Library oder
- Microsoft DAO 3.51 Object Library (oder höher)

Für Access 2000:

- Visual Basic For Applications
- Microsoft Access 9.0 Object Library (oder höher)
- Microsoft DAO 3.6 Object Library (oder höher)

Für Access 2002/XP:

- Visual Basic For Applications
- Microsoft Access 10.0 Object Library (oder höher)
- Microsoft DAO 3.6 Object Library (oder höher)



Beispiel für Access XP

Es können noch weitere Verweise aktiviert sein, aber die o.g. Verweise werden vom AP-Änderungsprotokoll-Manager in jedem Fall benötigt.

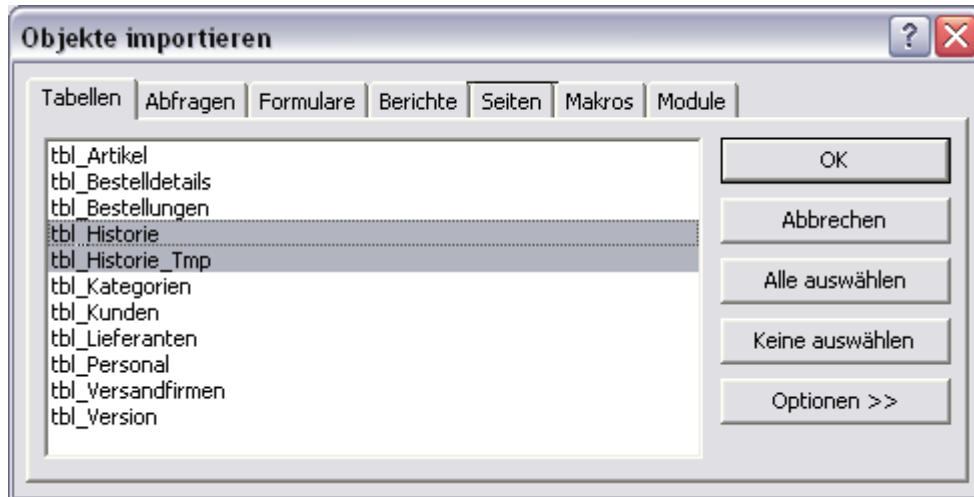
Zu importierende Objekte:

Folgende Objekte müssen aus der Originaldatei des AP-Änderungsprotokoll-Managers in Ihre Anwendung importiert werden:

Tabellen

tbl_Historie

tbl_Historie_Tmp



Formulare

frm_Historie

frm_Historie_Admin

frm_Historie_Admin_1

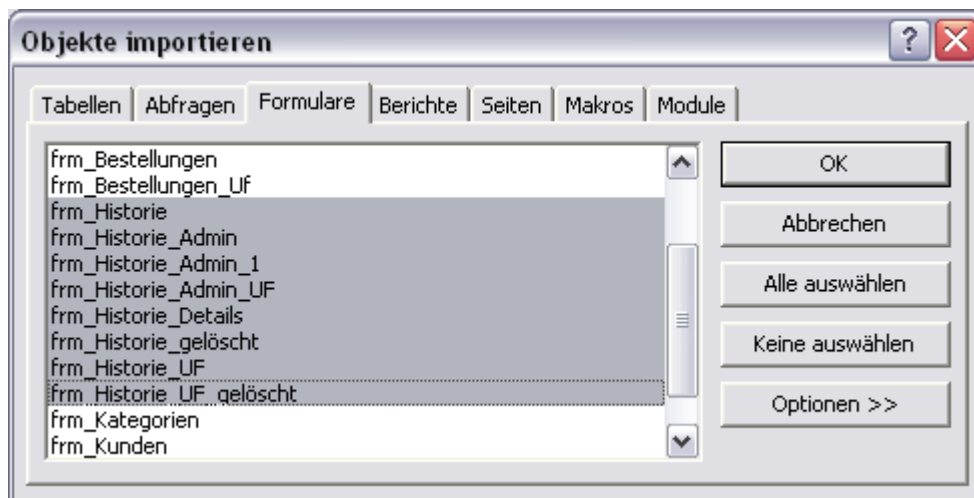
frm_Historie_Admin_UF

frm_Historie_Details

frm_Historie_gelöscht

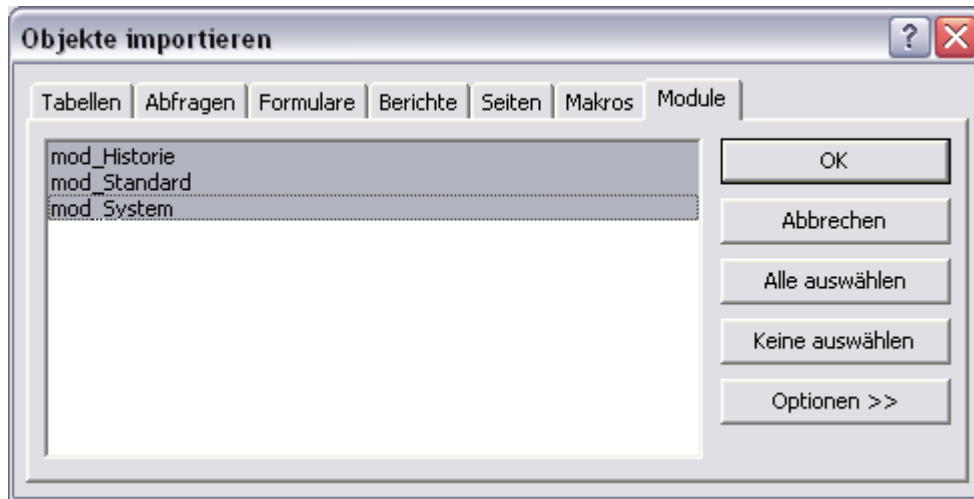
frm_Historie_UF

frm_Historie_UF_gelöscht



Module

mod_Historie
mod_Standard
mod_System



Hinweis:

Die folgenden Formulare sind nur in der **PRO**-Version des AP-Änderungsprotokoll-Managers vorhanden, da nur mit dieser Version Datensätze aus Unterformularen historisiert werden können.

frm_Historie_Admin_UF
frm_Historie_UF
frm_Historie_UF_gelöscht

Wie importiert man Objekte aus einer anderen Access-Datenbank?

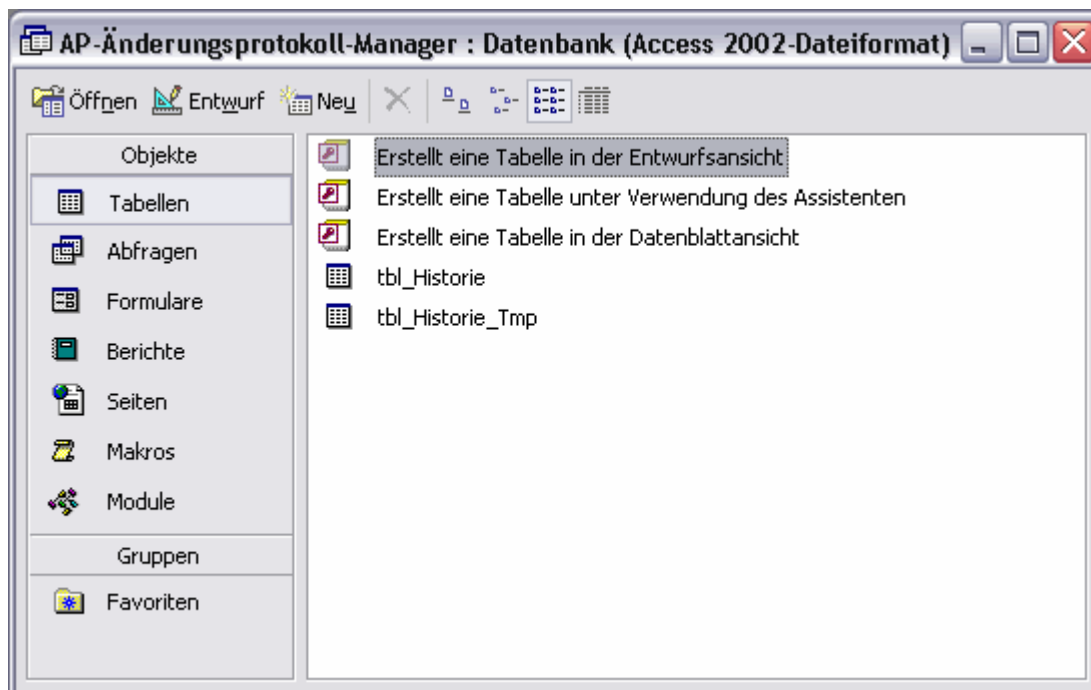
Öffnen Sie Ihre Anwendung. Über das Menü **Datei / Externe Daten / Importieren** wird der Dateidialog geöffnet. Wählen Sie über diesen Dialog die Access-Datei aus, aus der die Objekte importiert werden sollen. Es öffnet sich der Importdialog, den Sie in den oben abgebildeten Ansichten sehen. Hier können Sie auswählen, welche Tabellen, Abfragen usw. Sie in die aktuelle Datenbank importieren möchten.

Erklärung der importierten Module

Tabellen

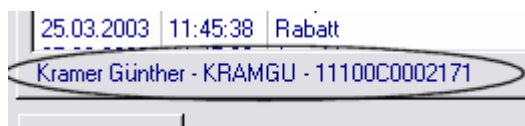
- tbl_Historie
 - Hier werden alle Daten dauerhaft historisiert und aufbewahrt

- tbl_Historie_Tmp
 - In dieser Tabelle werden die Daten vor dem Schließen des Formulars bzw. vor dem Löschen zwischengespeichert.
Wird das Speichern oder Löschen des Datensatzes abgebrochen, werden diese Daten nicht in die Tabelle tbl_Historie übertragen. Erst nach erfolgreicher Aktion (Speichern, Löschen oder Schließen des Formulars) werden aus dieser Tabelle die historisierten Daten in die eigentliche Historien-Tabelle übertragen. Die Tabelle tbl_Historie_Tmp dient nur der kurzzeitigen und temporären Zwischenspeicherung der Daten.



Formulare

- In fast allen Listen werden in der Statuszeile der Name des Access-Users, des Windows-User (Windowsanmeldung) und der Name des Rechners (Hostname) angezeigt. Alle Anzeigen und Listen lassen sich von erfahrenen Access-Entwicklern ändern und optisch anpassen. Sollten Sie eine andere Darstellung oder Sortierung der Spalten wünschen, ist dies kein Problem, da hierzu nur die SQL-Anweisung der Listen geändert werden muss.



- frm_Historie / frm_Historie_UF
 - In diesem Formular werden die Änderungen aufgelistet. Neben dem Datum und der Uhrzeit werden der Feldname, der alte Feldinhalt und der neue Inhalt des Feldes dargestellt.

Datum	Zeit	Feld	AlterWert	NeuerWert
25.03.2003	11:56:33	Verkaufsberater_Listenfeld	Davolio Nancy	Dodsworth Anne
25.03.2003	11:56:33	Versandunternehmen	Speedy	United
25.03.2003	11:56:33	Versanddatum		31.03.2002
25.03.2003	11:56:33	Lieferdatum		28.03.2003
25.03.2003	11:56:33	Verkaufsberater	Davolio Nancy	Dodsworth Anne
25.03.2003	11:35:20	Alle Felder		Neuanlage

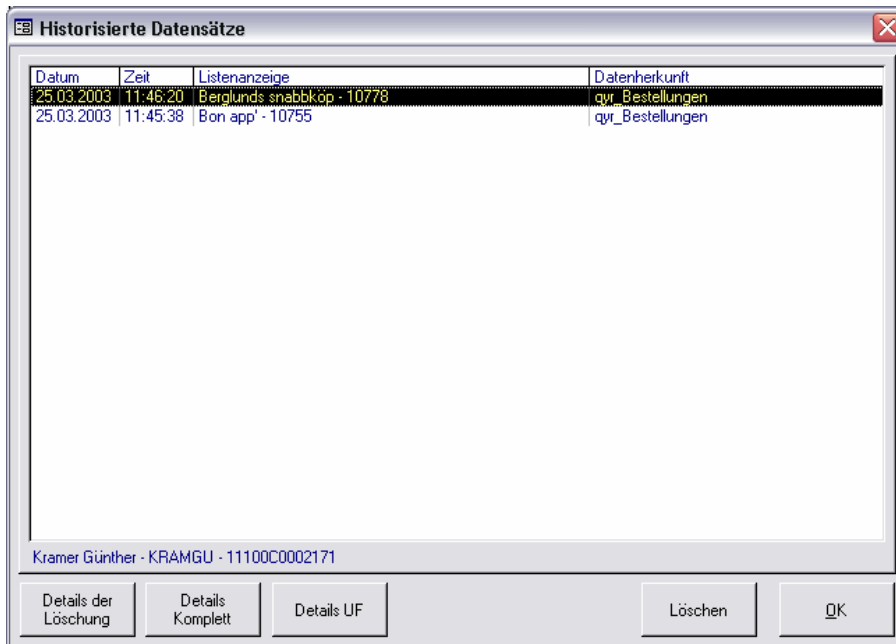
Kramer - F000217 - 77700C0002170

Die Schaltfläche **Details** öffnet das Formular **frm_Historie_Details**.

- frm_Historie_Details
 - Hier werden alle Infos zum protokollierten Datensatz angezeigt.

Geändert am:	
Datum:	<input type="text" value="25.03.2003"/>
Uhrzeit:	<input type="text" value="11:56:33"/>
Geändert von:	
Access-Anwender:	<input type="text" value="Kramer"/>
Windows-Anwender:	<input type="text" value="F000217"/>
Computername:	<input type="text" value="77700C0002170"/>
Infos der Änderung:	
Formularfeld:	<input type="text" value="Verkaufsberater"/>
Alter Feldinhalt:	<input type="text" value="Davolio Nancy"/>
Neuer Feldinhalt:	<input type="text" value="Dodsworth Anne"/>
Unterformular:	<input type="text"/>
Datenherkunft:	<input type="text" value="qyr_Bestellungen"/>
Datenherkunft (Art):	<input type="text" value="DATEN"/>
Datenherkunft (UF):	<input type="text"/>
<input type="button" value="OK"/>	

- frm_Historie_gelöscht
 - Bei komplett gelöschten Datensätzen muss mehr angezeigt werden, als der alte und neue Feldinhalt. Um die Daten den früheren Datensätzen zuzuordnen wird daher neben dem Daten und der Uhrzeit auch die Bezeichnung des Datensatzes und die Datenherkunft angezeigt. Die Datenherkunft ist entweder die wirkliche Datenherkunft (RecordSource) und gibt den Namen der Tabelle oder Abfrage wieder oder Sie geben diese durch eine aussagekräftige Bezeichnung vor. Im Beispielformular **frm_Artikel** haben wir diese Datenherkunft mit dem Text **Artikel** bezeichnet. Die Bezeichnung des Datensatzes legen Sie innerhalb des Formulars fest. Hier können Sie angeben, welche Inhalte aus dem Datensatz für die Listenanzeige verwendet werden sollen. Bei einem Datensatz für Artikel reicht in aller Regel die Artikelbezeichnung aus, da diese eindeutig ist. Haben Sie jedoch mehrere Rechnungen eines Kunden, sollten Sie neben der Kundenbezeichnung z.B. auch noch die Bestell-Nr. oder Rechnung-Nr. hinzufügen. Die Listenanzeige lässt sich aus mehreren Feldern zusammensetzen.

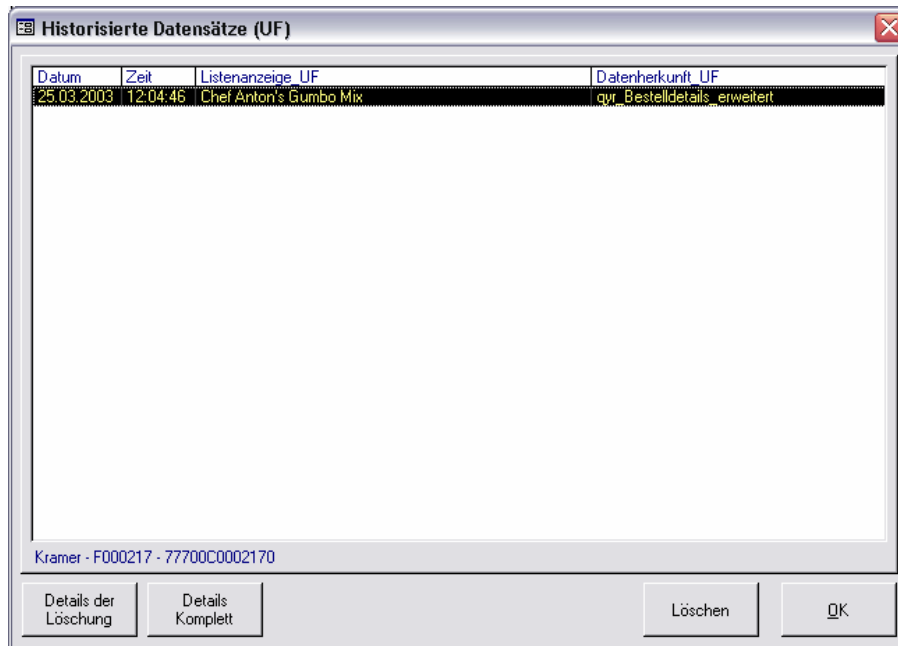


Die Schaltfläche **Details der Löschung** öffnet das Formular **frm_Historie** und zeigt nur die Inhalte zum Zeitpunkt der Datensatzlöschung an. Die Schaltfläche **Details Komplett** öffnet ebenfalls das o.g. Formular, zeigt hier aber alle Änderungen des Datensatzes an. Sollten angehängte Datensätze von Unterformularen vorhanden sein, werden diese durch Klicken von **Details UF** angezeigt. Hier öffnet sich dann das Formular **frm_Historie_UF_gelöscht**.

Möchten Sie den Datensatz mit allen protokollierten Daten aus der Historisierung entfernen, wählen Sie den gewünschten Datensatz an und klicken auf **Löschen**. Hierbei werden auch alle historisierten Daten aus den angehängten Unterformularen gelöscht.

- frm_Historie_UF_gelöscht

- Wie im Formular frm_Historie_gelöscht werden hier die gelöschten Daten angezeigt. Jedoch beschränkt sich diese Anzeige auf die Daten aus Unterformularen. Daher wird neben Datum, Uhrzeit und Listenanzeige des UFs auch die Datenherkunft des Datensatzes angezeigt. Diese kann aus der wirklichen Datenherkunft (RecordSource) oder aus einer eigenen Bezeichnung bestehen. Da in einem Hauptformular mehrere UFs mit verschiedenen Datenherkünften vorhanden sein können, kann der Betrachter sofort erkennen, in welchem Bereich der Anwendung dieser Datensatz früher angesiedelt war.



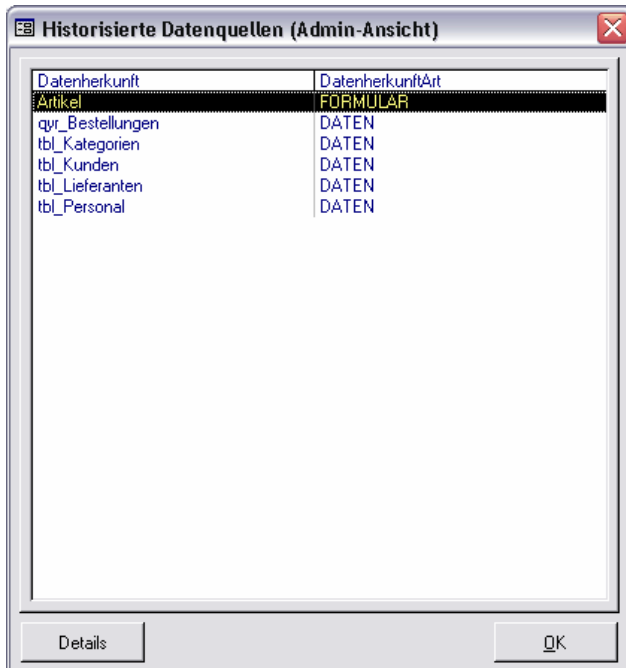
Die Schaltflächen **Details der Löschung** und **Details Komplet** rufen wie zuvor beschrieben das Formular **frm_Historie** mit dem entsprechenden Datenfilter auf. Durch das **Löschen** wird nur der jeweilige Datensatz mit allen dazugehörigen Daten gelöscht. Andere Datensätze dieses Hauptsatzes bleiben davon unbetroffen.

Administrationsansicht

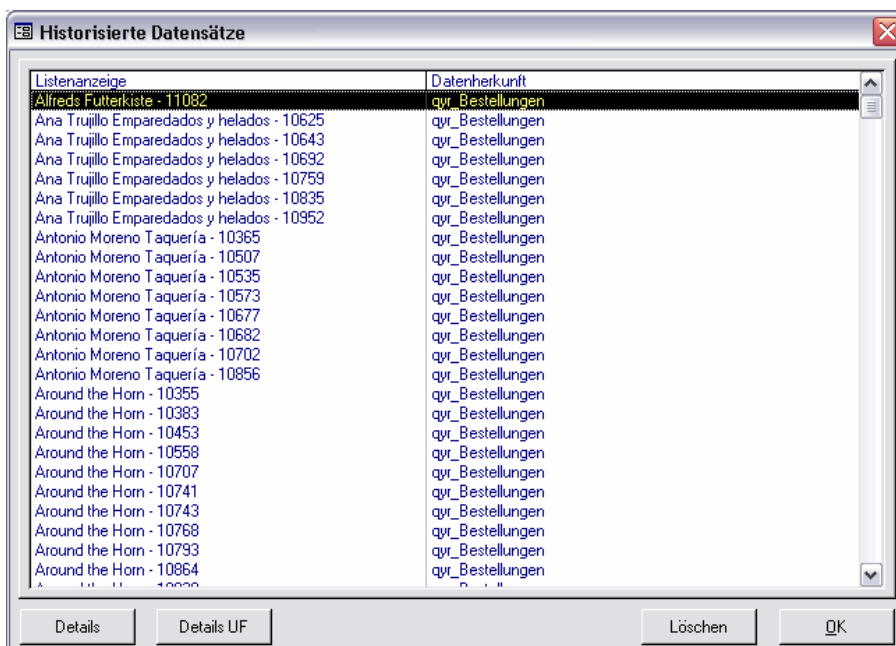
Die Anzeige der Historisierungen und Löschungen in den entsprechenden Formularen (Artikel, Kunden, Personal, Bestellungen usw.) ist in manchen Firmen und Unternehmen evtl. gar nicht gewünscht und wird daher über diesen Weg für den normalen Anwender nicht angeboten. Man kann den Aufruf für die Anzeige der Historisierungsdetails entweder vom angemeldeten Anwender abhängig machen oder die Anwendungsadministration verwendet die Adminansicht des AP-Änderungsprotokoll-Managers.

- frm_Historie_Admin

- Die Anzeige bietet die Auswahl aller Bereiche, für die historisierte Datensätze vorhanden sind. Die Schaltfläche **Details** öffnet das Formular **frm_Historie_Admin_1**.

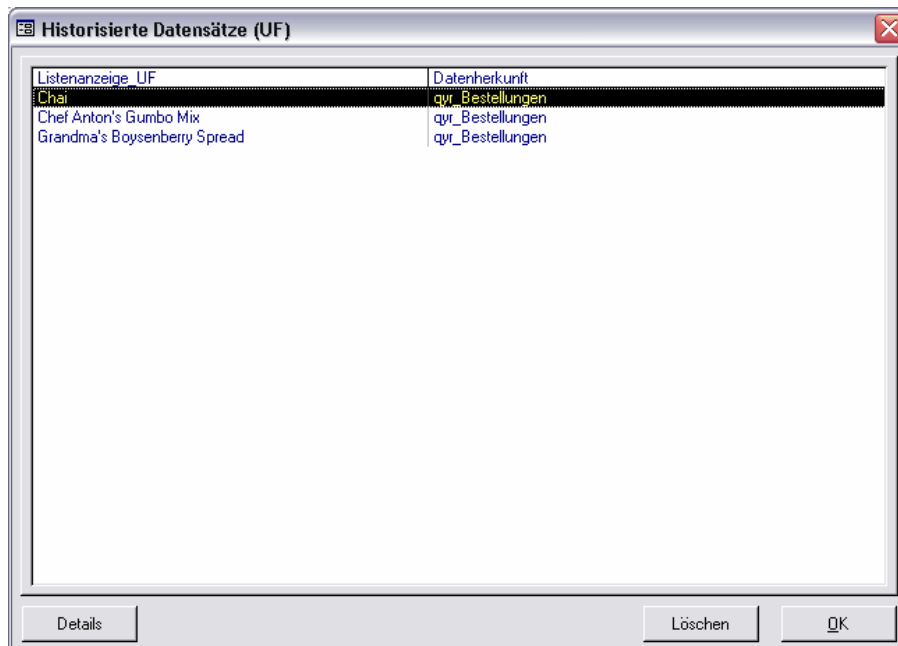


- frm_Historie_Admin_1
 - In dieser Ansicht werden alle Datensätze aus dem zuvor ausgewählten Bereich (frm_Historie_Admin) angezeigt.



Die Schaltfläche **Details** öffnet das Formular **frm_Historie**, **Details UF** ruft die Maske **frm_Historie_Admin_UF** auf.

- frm_Historie_Admin_UF
 - In diesem Formular werden alle historisierten Datensätze aus den angehängten Unterformularen gezeigt.



Die Schaltfläche **Details** öffnet das Formular **frm_Historie**.

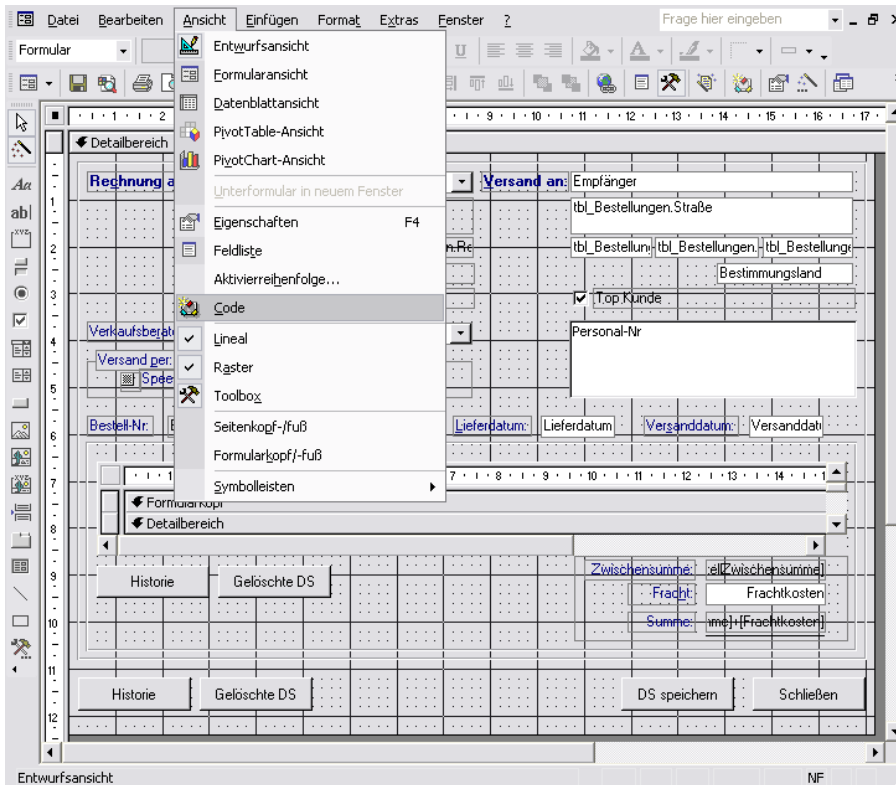
Module

- mod_Historie
In diesem Modul befindet sich die gesamte Logik der Historisierung und damit des AP-Änderungsprotokoll-Managers. In diesem Modul muss nichts geändert oder angepasst werden.
- mod_Standard
Einzig eine Funktion, die prüft ob ein angegebenes Formular geöffnet ist, befindet sich in diesem Modul. Haben Sie in Ihrer Anwendung bereits eine Funktion mit dem Namen **IstGeladen**, die ebenfalls diese Prüfung vornimmt, benötigen Sie dieses Modul nicht.
- mod_System
Die Funktionen dieses Moduls liefern den Windows-Anmeldenamen und den Computernamen bzw. den Hostnamen. Diese Informationen werden für die Historisierung verwendet. Wünschen Sie kein Speichern dieser Daten, kann der AP-Änderungsprotokoll-Manager dahingehend abgeändert werden.

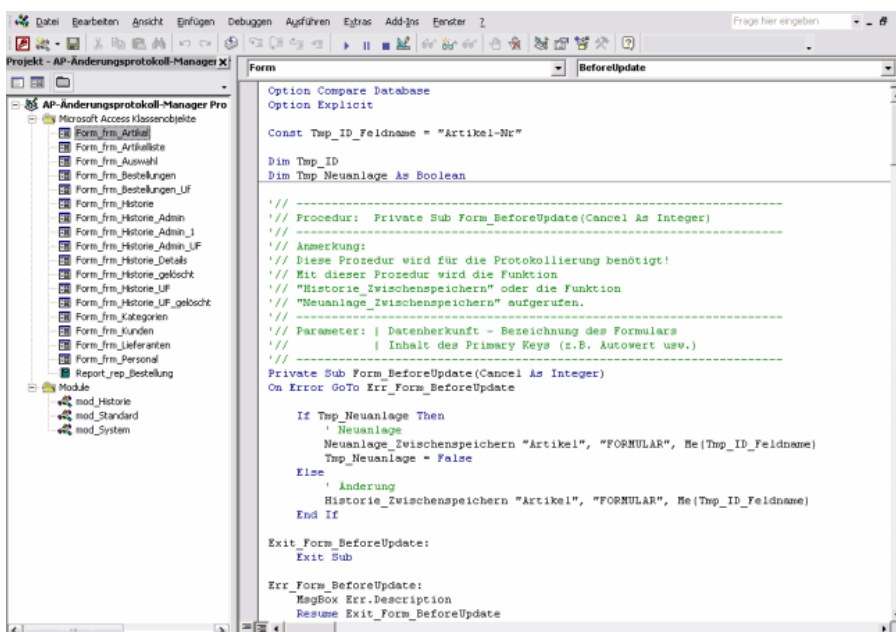
Anpassung Ihrer Formulare

In diesem Teil der Integrationsdokumentation erfahren Sie, wie Sie den VBA-Code in Ihre Masken einbauen und die Eigenschaften anpassen, damit die Historisierung Ihrer Daten erfolgen kann.

Öffnen Sie ein Formular, in welchem Sie die Neuanlagen, Änderungen und Löschungen protokollieren möchten, in der Entwurfsansicht. Wählen Sie aus dem Menü **Ansicht** den Punkt **Code** aus. Es öffnet sich die Codeansicht des Formulars.



Entwurfsansicht eines Formulars (frm_Bestellungen)



Codeansicht

Bewegen Sie den Cursor an den Anfang des Moduls und fügen Sie dort nach den Anweisungen **Option Compare Database** und **Option Explicit** folgenden Code ein:

```
Const Tmp_ID_Feldname = "Kunden-Code"  
Const DatenherkunftArt = "DATEN"
```

```
Dim Tmp_ID  
Dim Tmp_Neuanlage As Boolean
```

Wie Eingangs bereits erwähnt, benötigen Sie für die Historisierung der Daten einen eindeutigen Datensatzschlüssel (Autowert oder einen anderen Primary Key). Der Feldname des Schlüssels muss anstatt dem Text `Kunden-Code` eingefügt werden. Lautet der Primary Key Ihres Formulars bzw. der zur grundlegenden Datenherkunft z.B. `Artikel-Nr`, muss die Anweisung wie folgt lauten:

```
Const Tmp_ID_Feldname = " Artikel-Nr"
```

Den Inhalt für die Variabel `DatenherkunftArt` können Sie an dieser Stelle ebenfalls selbst bestimmen. Der Inhalt soll dazu dienen, bei verschiedenen Formularen mit gleicher Datenherkunft, unterscheiden zu können, in welchem Formular die Daten geändert wurden. Sie können hier einen beliebigen Text eingeben.

Die Variablen `Tmp_ID` und `Tmp_Neuanlage` werden zum Zwischenspeichern von Daten verwendet.

Die nachfolgenden Funktionen und Prozeduren kopieren Sie bitte aus der mitgelieferten Textdatei **Code für Formulare.txt** und fügen den Text in den Codebereich Ihres Formulars ein. Wir erklären welche Funktion wann angerufen wird. Es kann sein, dass es bereits in Ihrem Formular eine gleichnamige Funktion gibt. In diesem Fall müssen Sie den Funktionsinhalt (Code) aus der eingefügten Funktion in die vorhandene Funktion kopieren und den Rest (Funktionsanweisung) löschen. Wir gehen davon aus, sollten Sie bereits eine gleichnamige Funktion in Ihrem Formular haben, Sie dann auch in der Lage sind, die beiden Funktionen zusammenzuführen.

Beim Anzeigen:

Form_Current

Hier wird geprüft, ob es sich um einen neuen Datensatz handelt, damit bei der späteren Historisierung entschieden werden kann, ob alle geänderten Daten oder nur der Hinweis auf eine Neuanlage gespeichert werden soll. Für die Speicherung der Information wird die zuvor angelegte Variable `Tmp_Neuanlage` verwendet.

Vor Aktualisierung:

Form_BeforeUpdate

An dieser Stelle wird entschieden, ob es eine Neuanlage eines Datensatzes ist oder eine normale Datenänderung. Entsprechend des Inhalts der Variable `Tmp_Neuanlage` wird entweder die Funktion `Neuanlage_Zwischenspeichern` oder `Historie_Zwischenspeichern` aufgerufen. In beiden Fällen werden die Informationen zunächst in der Tabelle **tbl_Historie_Tmp** zwischengespeichert.

Nach Aktualisierung:

Form_AfterUpdate

Nach dem Schließen des Formulars bzw. Verlassen dieses Datensatzes werden die Daten aus der Tabelle **tbl_Historie_Tmp** in die Tabelle **tbl_Historie** übertragen.

Möchten Sie die komplette Löschung von Datensätzen historisieren, benötigen Sie noch die beiden folgenden Funktionen:

Beim Löschen:

Form_Delete

Diese Funktion wird bei einem Löschvorgang aufgerufen und speichert alle Daten des Datensatzes zwischen. Dazu wird die Funktion `Löschung_Zwischenspeichern` aufgerufen. In dieser Funktion werden auch bei der **PRO**-Version alle angehängten Datensätze aus den Unterformularen historisiert.

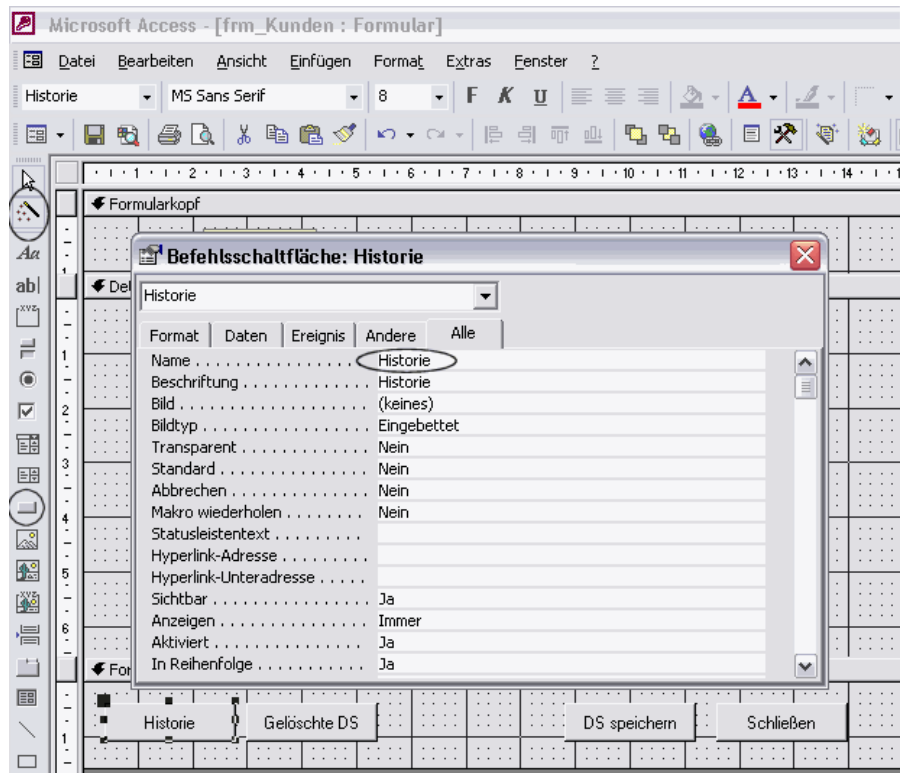
Nach Löschbestätigung:

Form_AfterDelConfirm

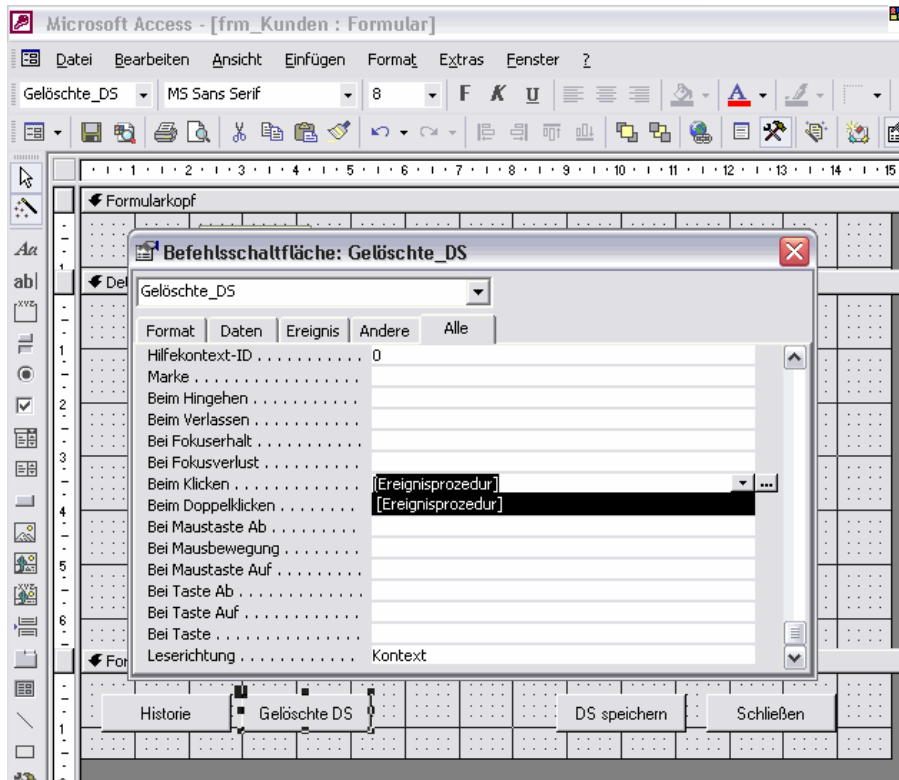
Wird diese Funktion aufgerufen, ist die Löschung des Datensatzes bestätigt bzw. abgeschlossen. Nun können die zwischengespeicherten Daten wie in der Funktion `Form_AfterUpdate` (Nach Aktualisierung) in die Tabelle **tbl_Historie** übertragen werden.

Soll in diesem Formular die Möglichkeit bestehen, von hieraus die historisierten Daten einzusehen, müssen zwei Schaltflächen angelegt werden. Haben Sie noch keine Schaltflächen in einem Access-Formular erstellt, hilft Ihnen die Access-Onlinehilfe dabei. Suchen Sie dort nach dem Begriff **Befehlsschaltflächen**.

Sie benötigen eine Schaltfläche für die Anzeige der Datenänderungen und eine für die Anzeige der gelöschten Datensätze.



Die Befehlsschaltfläche für die Anzeige der Änderungen, muss den Namen **Historie** tragen. Die Beschriftung kann frei gewählt werden. Die Schaltfläche, die alle gelöschten Datensätze anzeigt muss **Gelöschte_DS** heißen. Tragen Sie diesen Text als Namen für diese Befehlsschaltfläche ein.



Haben Sie die Schaltflächen erstellt, fügen Sie bitte aus der zuvor erwähnten Textdatei den Code für die folgenden Funktionen ein:

Historie – Beim Klicken:

`Historie_Click`

Gelöschte DS – Beim Klicken:

`Gelöschte_DS_Click`

In beiden Funktionen wird zunächst geprüft, ob bereits Daten für das Anzeigen vorhanden sind. Ist das der Fall, werden die Formulare **frm_Historie** bzw. **frm_Historie_gelöscht** aufgerufen und die gefilterten Daten für diesen Datensatz bzw. diese Datenherkunft angezeigt.

Klicken Sie in der Entwurfsansicht Ihres Formulars auf die eben angelegten Schaltflächen, öffnen die Ansicht der Eigenschaften über das Menü Ansicht / Eigenschaften, scrollen im Bereich Alle (Karteireiter Alle) fast bis an das Ende und klicken Sie mit einem Doppelklick in das Feld **Beim Klicken**. Es erscheint, sofern nicht bereits vorhanden, der Text **[Ereignisprozedur]**. Sie können auch mit einem einfachen Klick in das Feld, diesen Text zur Anzeige bringen. Dazu muss im Anschluss auf die Pfeilschaltfläche geklickt und der o.g. Text ausgewählt werden. Nun sind die zuvor eingefügten Funktionen `Historie_Click` und `Gelöschte_DS_Click` mit den Befehlsschaltflächen verbunden. Wenn in der Normalansicht des Formulars auf diese Schaltflächen geklickt wird, werden die Funktionen ausgeführt und die historisierten Daten in den entsprechenden Masken angezeigt.

In manchen Funktionen finden Sie folgende Codezeilen:

```
' Diese Information müssen Sie ggf. ändern (z.B. "Kunden")  
Datenherkunft = Me.RecordSource  
'Datenherkunft = "Kunden"
```

Möchten Sie anstatt des Namens der Tabelle bzw. Abfrage eine eigene Bezeichnung für die Datenherkunft haben, müssen Sie in allen Funktionen dieses Formulars die Bezeichnung in dem oben dargestellten Code austauschen. Statt `Me.RecordSource` fügen Sie dort einen Text Ihrer Wahl in Hochkommas ein.

Wichtige Anmerkung für Änderungen am Code

Verwenden Sie in den Bezeichnungen keine Hochkommas (einfach bzw. doppelt) und keine Kommas bzw. Semikolons.

Falsch:

```
"Kund'en"  
"Kund"en"  
"Kund, en"  
"Kund; en"
```

Hinweise für die PRO-Version und das Historisieren von Unterformularen

Nur in der **PRO**-Version ist das Historisieren von Datensätzen aus Unterformularen möglich. Wird ein Hauptdatensatz gelöscht, werden dabei auch die angehängten Datensätze aus den Unterformularen gesichert.

Um die Neuanlagen, Änderungen und Löschungen innerhalb eines Unterformulars zu historisieren, benötigen Sie noch weitere Funktionen, die direkt in den Codebereich des Unterformulars eingefügt werden müssen.

Den kompletten Code eines Unterformulars können Sie im Beispielformular **frm_Bestellungen_UF** einsehen.

Als zusätzliche Variable muss folgende Anweisung eingefügt werden:

```
Const Tmp_ID_Feldname2 = " Artikel-Nr"
```

Der Text **Artikel-Nr** ist mit dem Feldnamen des eindeutigen Datensatzschlüssels Ihres Unterformulars auszutauschen.

Der VBA-Code folgender Funktionen ist der selbe wie im Hauptformular:

Beim Anzeigen:	<code>Form_Current</code>
Nach Aktualisierung:	<code>Form_AfterUpdate</code>
Nach Löschestätigung:	<code>Form_AfterDelConfirm</code>

Der Code der folgenden Funktion ist für Unterformulare anders:

Vor Aktualisierung:	<code>Form_BeforeUpdate</code>
Beim Löschen:	<code>Form_Delete</code>

Den gesamten Code für ein Unterformular finden Sie in der Textdatei **Code für Unterformulare.txt**.

Anpassung Ihrer Masken und die darin enthaltenen Objekte

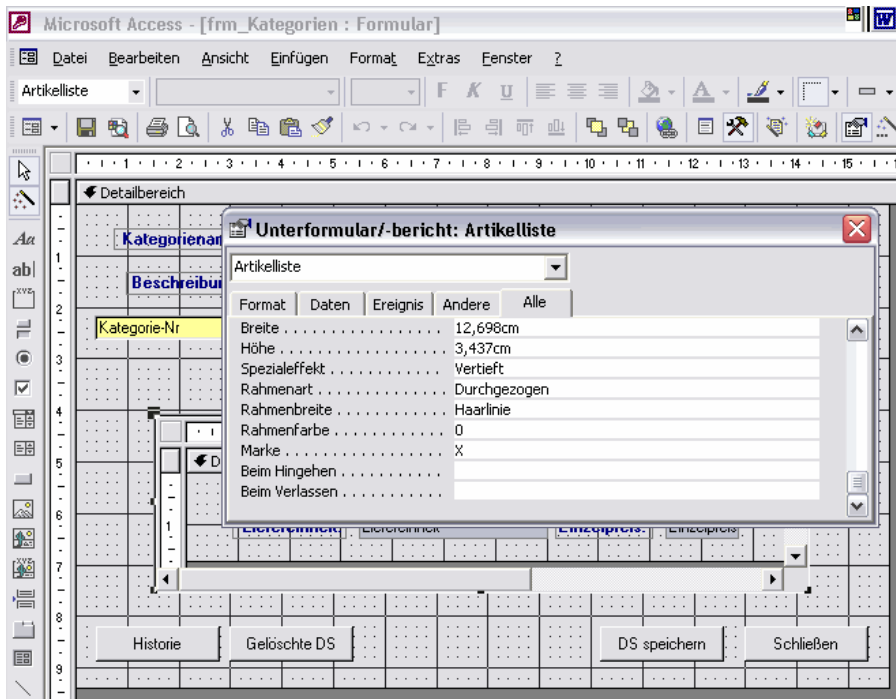
(Textfeld, Optionsgruppe, Kontrollkästchen, Listenfeld, Kombinationsfeld, Unterformular)

Zunächst sei angemerkt, dass nur Objekte aus der o.g. Aufzählung historisiert werden. Es macht wenig Sinn, den Inhalt eines Bezeichnungsfeldes oder einer Befehlsschaltfläche zu historisieren.

Für die Historisierung und die Ermittlung der genauen Feldinhalte müssen bestimmte Angaben und Parameter an die entsprechenden Funktionen übergeben werden. Wird z.B. ein Datensatz gelöscht, wird für die spätere Anzeige ein Listenfeld mit der Datensatzbezeichnung gefüllt. Das Modul muss wissen, welche Inhalte aus welchem Feld dafür verwendet werden sollen. Sie legen diese Felder und Informationen in den Eigenschaften bestimmter Objekte fest. Für einen Großteil verwenden wir dazu die Eigenschaft **Marke** bzw. **Tag** im Englischen. Da diese Eigenschaft in sehr wenigen Fällen verwendet wird, ist dieses Feld frei und kann mit Informationen für den AP-Änderungsprotokoll-Manager gefüllt werden.

Ausschluss eines Feldes bzw. Objektes für die Historisierung

Möchten Sie, dass der Inhalt eines bestimmten Feldes nicht historisiert wird, setzen Sie einfach ein großes X in die Marke-Eigenschaft des jeweiligen Objektes.



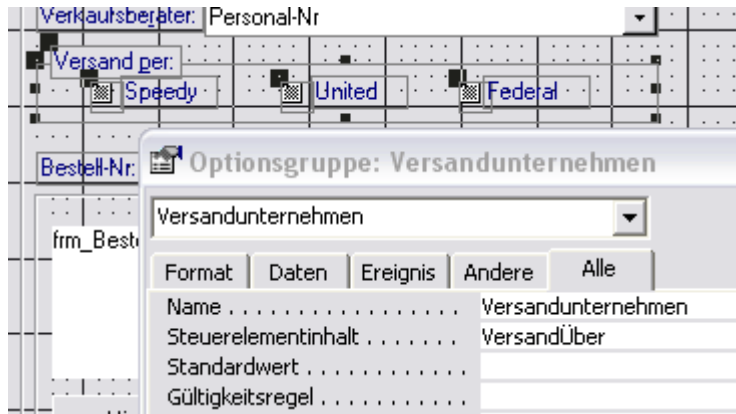
Im Beispielformular frm_Kategorien haben wir das Unterformular für die Protokollierung durch Setzen der Marke deaktiviert.

Textfeld

Der Inhalt eines Textfeldes wird so übernommen, wie dieser im Feld steht.

Optionsgruppe

Innerhalb einer Optionsgruppe gibt es verschiedene Inhalte, die aktiviert sein können. Diese Inhalte werden über einen Optionswert gesteuert. Dieser Wert ist eine Zahl und für die Historisierung nicht sehr aussagekräftig. Daher historisieren wir den Text, der an diesem Optionsfeld bzw. Kontrollkästchen hängt. Der Name des jeweiligen Bezeichnungsfeldes muss jedoch eine bestimmte Form haben, damit die Historisierung erfolgen kann.



Wenn der Namen der Optionsgruppe z.B. **Versandunternehmen** lautet muss der Name des Bezeichnungsfeldes für den ersten Wert **Versandunternehmen_1** lauten. Je nach Optionswert hängen Sie an den Namen einfach einen Unterstrich und den jeweiligen Optionswert an.



Kontrollkästchen

Im Beispielformular **frm_Artikel** gibt es ein Kontrollkästchen. In der Marke-Eigenschaft des Feldes **Auslaufartikel** steht der Text **Ja/Nein**.

Ist das Feld aktiviert und wird nun deaktiviert, so wird als alter Wert **Ja** und als neuer Wert **Nein** eingetragen.

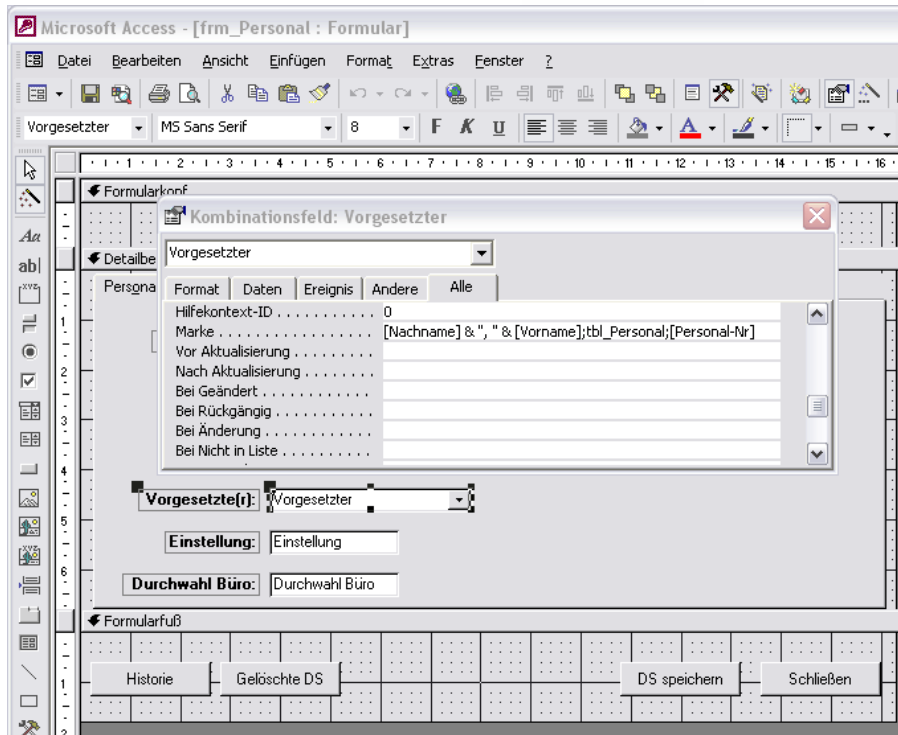
Möglichkeiten für die Marke-Eigenschaft sind:

- Ja/Nein
- Wahr/Falsch
- Ein/Aus



Listenfeld / Kombinationsfeld

Oft ist der Inhalt eines Feldes, welches über ein Listenfeld dargestellt wird, nur ein Autowert (Zahl) eines Wertes einer anderen Tabelle. Es macht keinen Sinn, diese Zahlen als historisierte Werte zu speichern. Vielmehr muss der innerhalb des Formulars angezeigte Wert historisiert werden. Das Listenfeld wird über eine Datenherkunft (Tabelle oder Abfrage) mit Daten versorgt. Bestimmte Informationen dieser Datenherkunft werden an die Funktionen des AP-Änderungsprotokoll-Manager übergeben. Diese ermittelt dann den eigentlichen Anzeigewert des Feldes.



Syntax

Um den eigentlichen Wert ermitteln zu können werden drei Parameter benötigt. Der erste Wert bezeichnet den Feldnamen bzw. die Feldnamen und liefert den Inhalt daraus zurück, welcher dann für dieses Listen-/Kombinationsfeld historisiert wird. Der zweite Parameter bezeichnet die Datenherkunft. Geben Sie hier die Tabelle oder Abfrage an.

Um aus der Datenherkunft den richtigen Datensatz ermitteln zu können, benötigt man einen Filter. Das Listen-/Kombinationsfeld hat einen Inhalt, der meistens eine Zahl eines Primary Keys oder eines Autowerts ist. Dieser Wert wird von der Funktion selbständig ermittelt. Was Sie im dritten Parameter angeben müssen ist das Feld, in dem dieser Wert gesucht werden soll.

Die Parameter müssen mit einem Semikolon getrennt in die Marke-Eigenschaft des Listen-Kombinationsfeldes eingetragen werden. Achten Sie darauf, dass zwischen den Parametern und den Semikolons kein Leerzeichen ist.

Im ersten Parameter können Sie mehrere Felder aus der Datenherkunft angeben. Sie finden hierzu ein Beispiel im Formular **frm_Personal** im Kombinationsfeld **Vorgesetzter**.

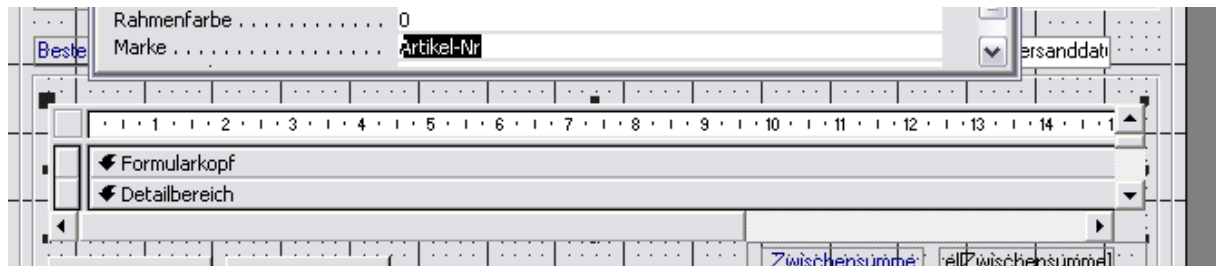
Im Beispielformular **frm_Kunden** gibt es ein Kombinationsfeld Land. Hier wird direkt der Inhalt, also der Ländername und kein Wert eines Schlüssels oder Autowerts, zurückgeliefert. In diesem Fall brauchen Sie keine Eintragungen in die Marke-Eigenschaft des Feldes vorzunehmen.

Weitere Beispiele für eine Wertermittlung mit Parameterübergabe finden Sie in den Formularen **frm_Artikel** und **frm_Bestellungen**. Im zuletzt genannten Formular haben wir aus Demonstrationszwecken für den Verkaufsberater zusätzlich ein Listenfeld eingefügt. Die Art der Parameterübergabe ist bei Kombinationsfeldern und Listenfeldern identisch.

Unterformular (nur bei der PRO-Version)

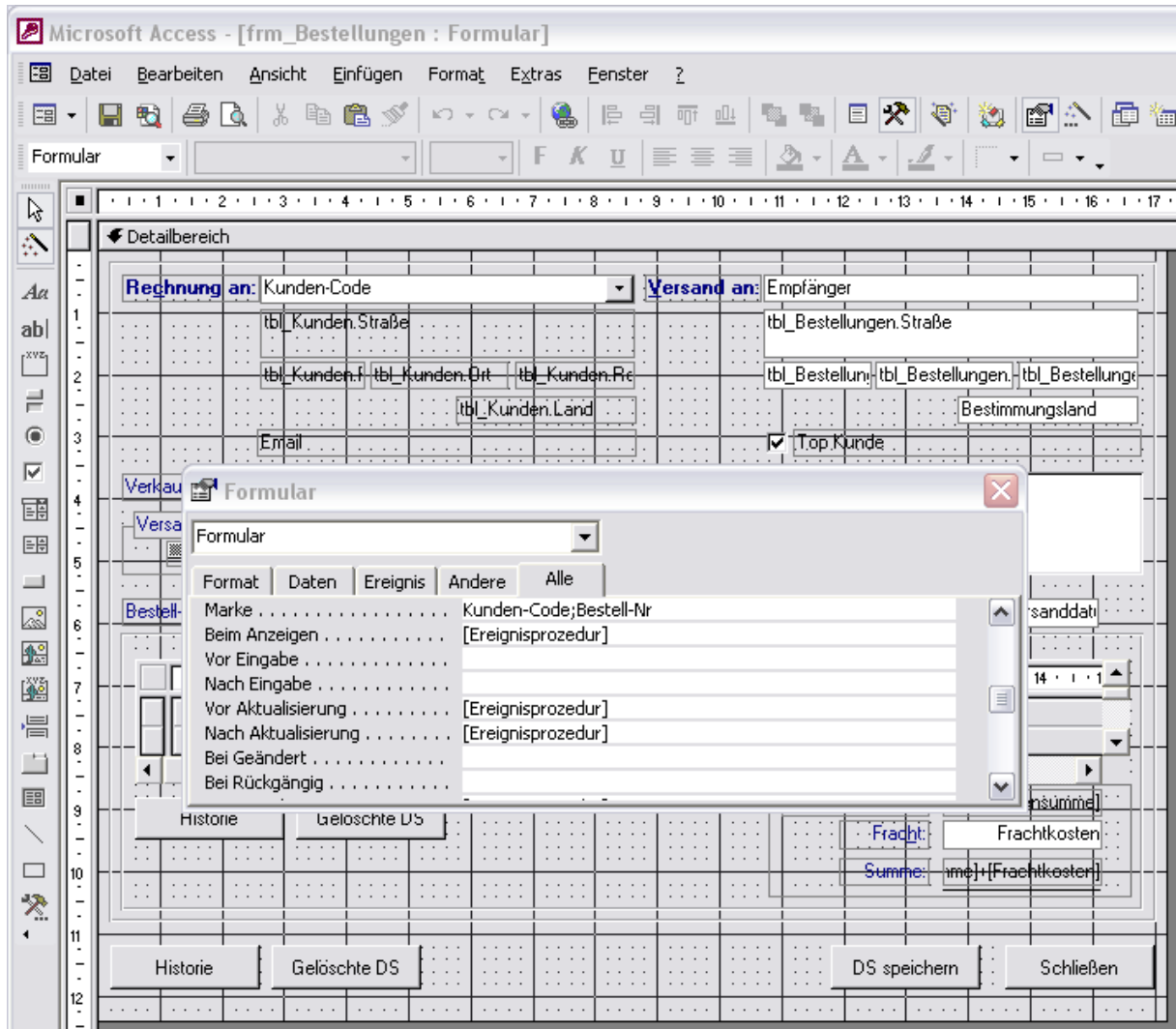
Verwenden Sie die Marke-Eigenschaften in Unterformularen genau wie es zuvor beschrieben wurde. Auch in Unterformularen lassen sich die eigentlichen Werte auf diese Art ermitteln.

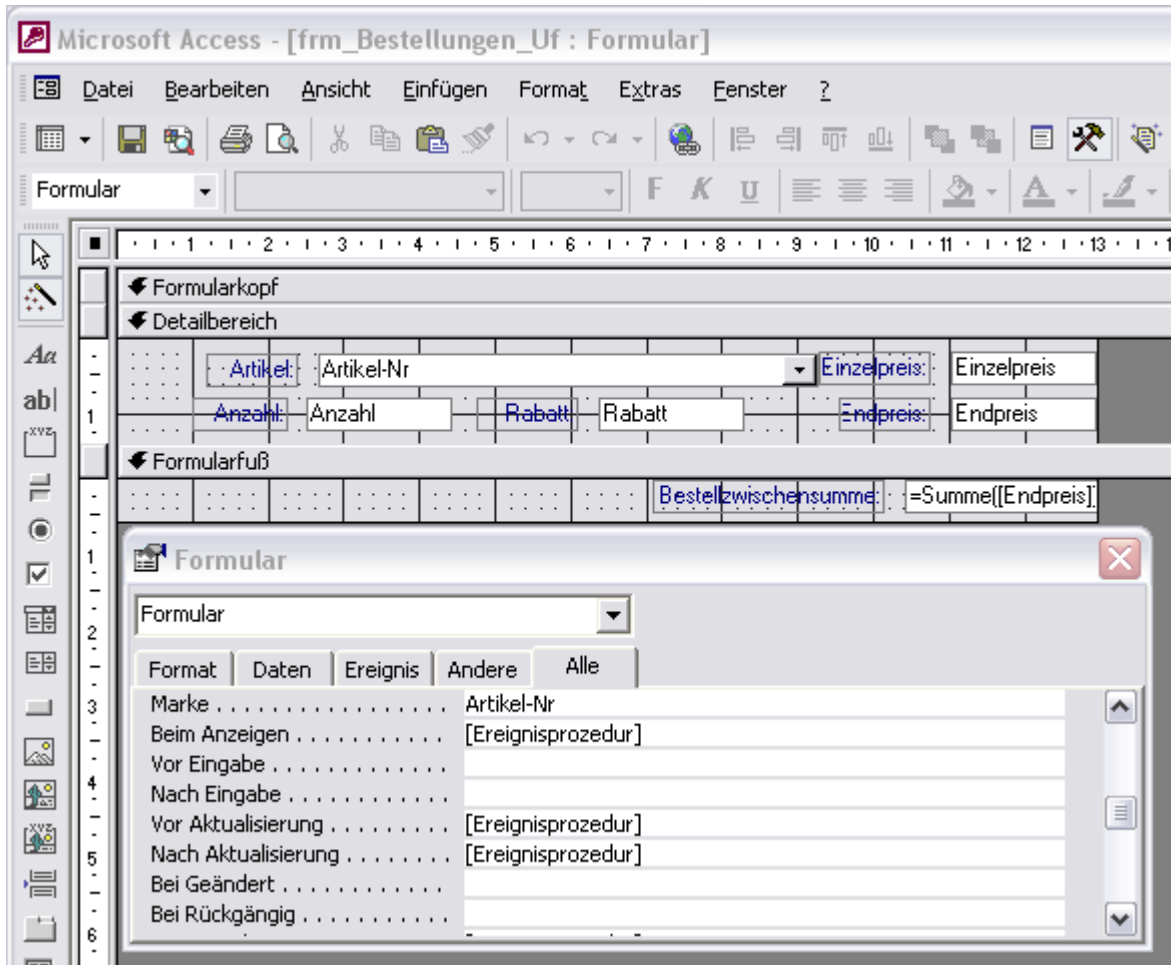
Damit in der Historisierung das Feld für die **ID_UF** mit den richtigen Daten gefüllt wird, müssen Sie im Unterformular-Objekt des Hauptformulars angeben, welchen eindeutigen Schlüssel es für die Datenherkunft des UFs gibt. In unserem Beispielformular **frm_Bestellungen** ist es für das Unterformular-Objekt **Bestellungen_Unterformular** das Feld **Artikel-Nr.**



Formular / Unterformular

Bei gelöschten Datensätzen oder Datensätzen aus Unterformularen wird eine Bezeichnung als Listenanzeige angezeigt. Diese wird aus Feldern innerhalb des Formulars bzw. Unterformulars ermittelt. Damit die entsprechende Funktion die Daten ermitteln kann, muss in der Marke-Eigenschaft des Formulars bzw. Unterformulars das Feld bzw. die Felder übergeben werden. Mehrere Feldnamen getrennt mit Semikolon sind hierbei möglich. Achten Sie darauf, dass zwischen den Feldnamen und dem Semikolon kein Leerzeichen ist. Bitte geben Sie die Feldnamen auch ohne Klammern ([und]) an. Diese werden von der Funktion selbst eingefügt.





Schlußwort:

Wir waren bei der Erstellung dieser Beschreibung bemüht, die Integration des AP-Änderungsprotokoll-Managers in Ihre Anwendungen so gut wie möglich zu beschreiben. Sollten Sie dennoch Fragen zum Einbau des Moduls haben, so schicken Sie diese per Mail an support@access-paradies.de. Teilen Sie uns in dem Mail bitte Ihre Kundennummer, Rechnungsnummer und die von Ihnen verwendete Accessversion mit.

Wir wünschen Ihnen viel Freude an dem AP-Änderungsprotokoll-Manager

Microsys Kramer
Access-Paradies

<http://www.access-paradies.de>
<http://www.ms-office-forum.net>