Die Datenzugriffsmethoden von Access im Praxiseinsatz

Teil 1  
Versionen: Access 2013, 2010, 2007, 2003 und 2002  
  
Für den programmatischen Zugriff auf eine Access-Datenbank wird die Datenzugriffsmethode "DAO" (Data Access Objects) oder "ADO" (ActiveX Data Objects) eingesetzt. Damit Sie diese Methoden nutzen können, muss in der VBA-Entwicklungsumgebung über das Menü *Extras-Verweise* eine Referenz auf das gewünschte Objektmodell eingerichtet sein. In diesem Zusammenhang gibt es allerdings immer wieder Verwirrungen und Fehlermeldungen, weil die Referenzen teils automatisch eingerichtet werden, teils aber auch manuell zu setzen sind und beide Datenzugriffsmethoden identisch benannte Objekte bereitstellen, die jedoch unterschiedlich aufgebaut sind. Unser Tipp bringt ein wenig Licht ins Dunkel und hilft Ihnen, Laufzeitfehler in Zukunft zu vermeiden.  
  
Sehen wir uns zunächst einmal an, wann Access automatisch einen Verweis auf eine Datenzugriffsmethode einrichtet: Das ist immer dann der Fall, wenn Sie eine neue Datenbank anlegen. Je nach Version richtet Access in Version 2003 und 2002 eine Referenz auf "Microsoft ActiveX Data Objects 2.x - 6.x Object Library" (ADO) ein. Soll stattdessen oder zusätzlich DAO genutzt werden, ist manuell eine Referenz auf "Microsoft DAO 3.6 Object Library" zu setzen und die Referenz auf ADO ggf. zu entfernen. Geschieht das nicht, besteht die Gefahr, dass Laufzeitfehler auftreten können, denn sowohl DAO als auch ADO stellen zum Beispiel ein Objekt "Recordset" zur Verfügung. Problematisch sind dann Anweisungen wie die folgenden:  
  
Dim rs As New Recordset  
rs.Open("Kunden")  
  
Als Datenzugriffsmethode verwendet VBA in diesem Fall das Objektmodell, das im Dialog "Verweise" als erstes referenziert ist. Sollte das DAO sein, kommt es zu Laufzeitfehlern, denn mit DAO lässt sich weder ein Recordset mit "New" instanziieren noch stellt es eine Methode "Open" zur Verfügung. Von der Lösung, nun einfach im Dialog "Verweise" die Priorität für ADO vor DAO zu setzen, ist dringend abzuraten, denn so erkennen Sie nie, welches Objektmodell in Ihrem Code aktuell verwendet wird. Gewöhnen Sie sich stattdessen an, die Objektmodellbezeichner "DAO" oder "ADODB" bei der Deklaration voranzustellen, so gibt es keine Missverständnisse und auch keine Laufzeitfehler mehr:  
  
Dim rsIn As New ADODB.Recordset  
Dim rsOut As DAO.Recordset  
  
Während der Objektmodellbezeichner bei "DAO" auch einfach nur "DAO" lautet, wird bei "ADO" der Bezeichner "ADODB" verwendet. Das resultiert daraus, dass es noch eine erweiterte ADO-Bibliothek namens "Microsoft ADO Ext 2.x|6.x for DDL and Security" mit dem Bezeichner "ADOX" gibt. "ADODB" referenziert die Standardobjekte und ADOX spezielle Objekte für DDL (Data Definition Language) und für sicherheitsrelevante Aufgaben.  
  
In Version 2013, 2010 und 2007 wird bei neuen Datenbanken eine Referenz auf "Microsoft Office x.x Access Database Engine Object Library" (x.x= 15.0 bei Access 2013, 14.0 bei Access 2010, 12.0 bei Access 2007) eingerichtet. Diese neue, mit Access 2007 eingeführte Database Engine - auch als ACE (Access Connectivity Engine) bekannt - enthält DAO. Das kann bei der Umsetzung von Tipps oder Schritt-für-Schritt-Anleitungen zu Problemen führen. Heißt es dort etwa "Richten Sie einen Verweis auf 'Microsoft DAO 3.6 Object Library' ein" kann es im Dialog "Verweise" nach Aktivierung des Kontrollkästchen und dem Klick auf *Ok* zu folgender Fehlermeldung kommen:  
  
Name steht im Konflikt mit vorhandenem Modul, Projekt oder vorhandener Objektbibliothek  
  
Da es bereits einen Verweis auf die "Microsoft Office x.x Access Database Engine Object Library" mit integrierter DAO-Bibliothek gibt, führt der Versuch, einen weiteren Verweis auf DAO einzurichten, zu der erwähnten Fehlermeldung. Sie können in diesen Fällen auf den Verweis auf die "Microsoft DAO 3.6 Object Library" verzichten.  
  
Eine Referenz auf eine Datenzugriffsmethode kann sich aber auch ändern, wenn Sie eine vorhandene Datenbank konvertieren. Wann genau das geschieht und was in diesen Fällen zu beachten ist, zeigen wir im zweiten Teil dieses Tipps.

Die Datenzugriffsmethoden von Access im Praxiseinsatz, Teil 2  
Versionen: Access 2013, 2010, 2007, 2003 und 2002  
  
Für den programmatischen Zugriff auf eine Access-Datenbank wird die Datenzugriffsmethode "DAO" (Data Access Objects) oder "ADO" (ActiveX Data Objects) eingesetzt. Damit Sie diese Methoden nutzen können, muss in der VBA-Entwicklungsumgebung über das Menü *Extras-Verweise* eine Referenz auf das gewünschte Objektmodell eingerichtet sein. In diesem Zusammenhang gibt es allerdings immer wieder Verwirrungen und Fehlermeldungen, weil die Referenzen teils automatisch eingerichtet werden, teils aber auch manuell zu setzen sind und beide Datenzugriffsmethoden identisch benannte Objekte bereitstellen, die jedoch unterschiedlich aufgebaut sind. Unser Tipp bringt ein wenig Licht ins Dunkel und hilft Ihnen, Laufzeitfehler in Zukunft zu vermeiden.  
  
Im ersten Teil dieses Tipps der Woche hatten wir gezeigt, wann Access automatisch einen Verweis auf eine Datenzugriffsmethode einrichtet und was zu beachten ist, wenn Sie Verweise auf eine andere oder eine zusätzlich zu verwendende Datenzugriffsmethode setzen. Eine Referenz auf eine Datenzugriffsmethode kann sich aber auch ändern, wenn Sie eine vorhandene Datenbank konvertieren. Wann genau dies geschieht und was in diesen Fällen zu beachten ist, wollen wir nun in diesem zweiten Teil des Tipps der Woche untersuchen.  
  
Bei Datenbanken, die in Access 97 angelegt wurden, war die standardmäßig eingerichtete Datenzugriffsmethode "DAO", während Access 2000, genau wie 2003 und 2002, einen Verweis auf ADO für neue Datenbanken eingerichtet hat. Bei der Konvertierung einer Access 97-Datenbank in die Version 2003 oder 2002 wird die DAO-Referenz automatisch übernommen bzw. aktualisiert, hier sind keine weiteren Aktionen erforderlich. Gleiches gilt für eine Access 2000-Datenbank, die Sie in Version 2003, oder 2002 konvertieren: Vorhandene DAO- oder ADO-Referenzen werden ebenfalls automatisch übernommen bzw. aktualisiert. Die Konvertierung einer Access 2002-Datenbank in eine Access 2003-Datenbank ist nicht erforderlich, da beide Versionen das gleiche Datenbankformat verwenden.  
  
Etwas spannender wird es, wenn Sie eine Access 97-Datenbank mit Access 2013, 2010 oder 2007 nutzen möchten: Während Sie Datenbanken aus Access 2003, 2002 oder 2000 in einem Kompatibilitätsmodus direkt nutzen können, müssen Sie eine Access 97-Datenbank zunächst zwingend in das neue Datenbankformat ACCDB konvertieren. Eine eventuell vorhandene Referenz auf "Microsoft DAO 3.6 Object Library" wird durch eine Referenz auf "Microsoft Office x.x Access Database Engine Object Library" (x.x= 15.0 bei Access 2013, 14.0 bei Access 2010, 12.0 bei Access 2007) ersetzt, so dass damit das in der ACE enthaltene DAO referenziert ist. Sollte die Access 97-Datenbank eine Referenz auf ADO enthalten, wird diese übernommen bzw. aktualisiert. Ähnliches gilt für die Konvertierung einer Access 2003-, 2002- oder 2000-Datenbank in das ACCDB-Format: Eine DAO-Referenz wird durch eine ACE-Referenz ersetzt, eine ADO-Referenz übernommen bzw. aktualisiert. Die Konvertierung einer Access 2007- oder 2010-Datenbank in eine neuere Version ist bei Desktop-Datenbanken nicht erforderlich, da alle Versionen das gleiche Datenbankformat nutzen.  
  
Vorsicht ist angesagt, wenn Sie sich beim Einsatz von DAO nach der Konvertierung entscheiden, die Referenz auf "Microsoft Office x.x Access Database Engine Object Library" zu deaktivieren und eine Referenz auf "Microsoft DAO 3.6 Object Library" zu verwenden: Die ACE enthält eine abwärtskompatible DAO-Bibliothek, mit der Sie vorhandene DAO-Routinen ohne Anpassungen nutzen können. Sie enthält aber auch Erweiterungen beispielsweise für den Zugriff auf Mehrfach- oder Anlagenfelder per Recordset, die Sie dann nicht mehr nutzen können!  
  
Außerdem ist zu beachten, dass die ACE sowohl 32- als auch 64-Bit-fähig ist, während die normale DAO-Bibliothek nur in einer 32 Bit-Variante vorliegt. Datenbanken, die DAO referenzieren, können somit nicht mit Access 2013 oder 2010 in der 64 Bit-Variante genutzt werden! ADO liegt ebenfalls in 32- und 64-Bit-Varianten vor; hier sind also ebenfalls keine besonderen Vorkehrungen erforderlich.