



Session D-SQL SQLization – endlich Client/Server!

Uwe Habermann
Uwe@hceood.eu

Einführung

Umstellung von VFP Anwendungen von der Verwendung von VFP-Datenbanken auf SQL Server-Datenbanken. Was ist bei der Umstellung zu beachten? Wie können lokale Ansichten und der programmatische Zugriff auf Daten angepasst werden? Die Verwendung von Cursoradaptern wird erläutert und es werden die Werkzeuge für das Upsizing und die Änderung des Datenzugriffs gezeigt. Die Beispiele basieren auf dem Quellcode und den Tools des dFUG-Frameworks Visual Extend.

Warum SQL Server?

Ein großer Kunde von uns entwickelt seit langer Zeit VFP Anwendungen, die auf einem selbst entwickelten Framework basieren. Der gesamte Datenzugriff ist in diesem Framework gekapselt.

Das Framework verwendet ausschließlich lokale Ansichten, um auf VFP Datenbanken zuzugreifen.

Schon bei der Entwicklung des Frameworks hat man daran gedacht, den Datenzugriff zu einem späteren Zeitpunkt auf die Verwendung anderer Datenbanken umzustellen. Man hat aber keinen Code entwickelt, der auf andere Datenbanken zugreifen könnte.

Nun hat man sich, inzwischen vor einigen Jahren, überlegt, wie der Datenzugriff auf eine SQL Server Datenbank gestaltet werden kann. Man ist zu dem Schluss gekommen, dass der Zugriff mit Cursoradaptern wesentlich flexibler und sicherer ist, als der Zugriff mit remote Ansichten.

Also wurde das Framework erweitert, so dass der Datenzugriff nicht nur mit lokalen Ansichten, sondern auch mit Cursoradaptern erfolgen kann. Über eine Eigenschaft des Frameworks, deren Wert aus einer Konfigurationsdatei gelesen wird, kann eingestellt werden, ob mit einer VFP Datenbank oder einer SQL Server Datenbank gearbeitet werden soll.

Dies ist nur ein Beispiel.

Tatsächlich werden immer mehr Anwendungen von der Verwendung von VFP Datenbanken auf SQL Server Datenbanken umgestellt. Diese Umstellung kann verschiedene Gründe haben.

Geschwindigkeit

Wenn der Datenzugriff gut optimiert ist, kann der Zugriff auf eine SQL Server Datenbank im Netzwerk tatsächlich wesentlich schneller sein, als der native Zugriff auf eine VFP Datenbank. Wenn der Datenzugriff nicht optimiert ist, kann man allerdings auch den gegenteiligen Effekt erreichen.

Datensicherungen

SQL Server erlauben die Durchführung von Backups während mit einer Datenbank gearbeitet wird. Solche Backups werden Online-Backups genannt. Der SQL Server stellt sicher, dass ein konsistenter Datenbestand gesichert wird.

Sicherheit

SQL Server-Datenbanken gelten als sicher. Es wird aus nur eine Anwendung, dem SQL Server selbst, in die Datenbank geschrieben und aus der Datenbank gelesen. Der SQL Server ist ein Dienst, der auf einem Windows Rechner ausgeführt wird. Der SQL Server kommuniziert mit den Clients und stellt sicher, dass nur vollständige, gültige Abfragen verarbeitet werden. Der direkte Zugriff auf eine SQL Server-Datenbank von mehreren Anwendungen oder über das Netzwerk ist nicht möglich. Schon allein hierdurch ist eine hohe Sicherheit gewährleistet.

Anders als in FoxPro, wo jede Tabelle, Indexdatei und Memodatei in jeweils einer physikalischen Datei abgelegt ist, ist beim SQL Server die gesamte Datenbank in einer einzigen physikalischen Datei. Diese Datei ist vom SQL Server-Dienst im ständigen Zugriff, so dass auf die Datenbank kein weiterer Zugriff möglich ist.

Skalierbarkeit

In FoxPro ist die Größe einer Datei auf 2 GB beschränkt. Beim SQL Server kann eine Datenbank bis zu 512 TB groß werden.

Bei den kostenlosen SQL Server Express Editionen hat Microsoft die Beschränkung der Datenbankgröße immer wieder gelockert.

Während früher die Datenbankgröße auf 2 GB begrenzt war, erlauben SQL Server 2005 Express, SQL Server 2008 Express und SQL Server 2008 R1 Express 4 GB als maximale Datenbankgröße.

Seit SQL Server 2008 R2 Express über SQL Server 2012 Express bis zur aktuellen Version SQL Server 2014 Express beträgt die maximale Datenbankgröße 10 GB.

Beim Upsizing einer vorhandenen VFP Datenbank zu einer SQL Server Datenbank ist also die Größe der Datenbank zu beachten, wenn eine SQL Server Express Version verwendet werden soll.

Der Zugriff von einer lokal laufenden VFP Anwendung auf eine VFP Datenbank erfolgt über das Netzwerk. VFP liest aus Indexdateien und Tabellen, so dass die gewünschten Daten angezeigt werden. Bei der Programmierung haben wir darauf zu achten, dass die Regeln der Rushmore-Optimierung eingehalten werden, damit der Datenzugriff optimiert ist.

Der Zugriff von einer VFP Anwendung auf eine SQL Server Datenbank erfolgt über einen ODBC-Treiber. Der ODBC-Treiber sendet die Abfrage an den SQL Server über das Netzwerk. Eine Abfrage besteht aus einer Zeichenfolge, die in der Regel einige hundert bis einige tausend Zeichen umfasst. Der SQL Server wird in der Regel auf der Maschine ausgeführt, auf der die Datenbankdatei gespeichert ist. Der eigentliche Datenbankzugriff erfolgt also nicht über das Netzwerk. Der SQL Server liefert an den ODBC-Treiber genau die gewünschten Daten zurück.

Durch dieses Zugriffsverfahren ist die Netzwerkbelastung sehr viel geringer, als beim Zugriff auf eine VFP Datenbank. Es können sehr viel mehr Benutzer auf eine SQL Server Datenbank gleichzeitig zugreifen.

Warum Cursoradapter?

Grundsätzlich bietet FoxPro drei Möglichkeiten um auf die Daten aus remote Datenbanken zuzugreifen.

Remote Ansichten

Die Verwendung von remote Ansichten steht uns seit VFP 3 zur Verfügung. Hiermit wird uns eine einfache Möglichkeit gegeben um auf remote Datenbanken zuzugreifen. Schließlich ist die Arbeit mit remote Ansichten praktisch genauso, wie die Arbeit mit lokalen Ansichten, mit der sich Entwickler schnell vertraut machen können.

Diese einfache Möglichkeit bringt aber Nachteile mit sich. Trotz der Verwendung einer remote Datenbank muss ein VFP Datenbank-Container vorhanden sein. Die remote Ansichten werden in einem Datenbank-Container gespeichert. Außerdem wird in dem Datenbank-Container die Verbindungszeichenfolge gespeichert und kann mit relativ einfachen Mitteln ausgelesen werden. Dies stellt ein Sicherheitsrisiko dar.

SPT – SQL Pass Through

Bei SPT werden SQL Befehle programmatisch erstellt und direkt an den ODBC-Treiber weitergereicht. Dieses Verfahren ist sehr schnell. Die Verbindungszeichenfolge kann ebenfalls im Programmcode abgelegt werden. Damit ist die Verwendung von SPT nicht nur schnell, sondern auch sicher.

Nachteil ist der große Programmieraufwand.

Cursoradapter

Cursoradapter sind VFP Basisklassen. Alle Vorteile und Möglichkeiten der objektorientierten Programmierung können bei Cursoradaptern angewendet werden. Cursoradapter haben also Eigenschaften und Methoden und können abgeleitet werden.

Cursoradapter können als Datenquelle in der Datenumgebung von Formularen verwendet werden.

Umstellung ausgehend von nativem Tabellenzugriff

Beim direkten Zugriff auf VFP Tabellen, befinden sich diese in der Regel in der Datenumgebung von Formularen. Der Alias entspricht dem Tabellennamen. Damit enthalten auch die Controlsources der Steuerelemente auf Formularen den Tabellennamen.

Beim programmatischen Zugriff auf VFP Tabellen wird meistens ebenfalls der Tabellename als Alias verwendet.

Hierdurch wird die Umstellung des Datenzugriffs auf Cursoradapter erschwert. Cursoradapter müssen die verwendete Tabelle intern öffnen. Hierbei wird die Tabelle grundsätzlich mit dem Tabellennamen als Alias geöffnet. Der Cursor des Cursoradapters muss einen anderen Aliasnamen bekommen. Es ist also nicht einfach möglich in der Datenumgebung von Formularen die Tabellen durch Cursoradapter zu ersetzen. Es müssten auch die Controlsources von allen Steuerelementen geändert werden, so dass der Aliasname des Cursors verwendet wird. Auch beim programmatischen Datenzugriff müssten alle Aliasnamen geändert werden. Der Änderungsaufwand wäre also groß.

Umstellung ausgehend von lokalen Ansichten

Wenn der bisherige Datenzugriff ausschließlich auf lokalen Ansichten basiert, ist die Umstellung auf remote Ansichten relativ einfach.

Es muss ein neuer Datenbank-Container erstellt werden, der eine Verbindung zum SQL Server enthält sowie alle Ansichten. Die bisherigen lokalen Ansichten müssen durch remote Ansichten ersetzt werden. Die Syntax des SELECT Befehls kann hierbei oft unverändert bleiben.

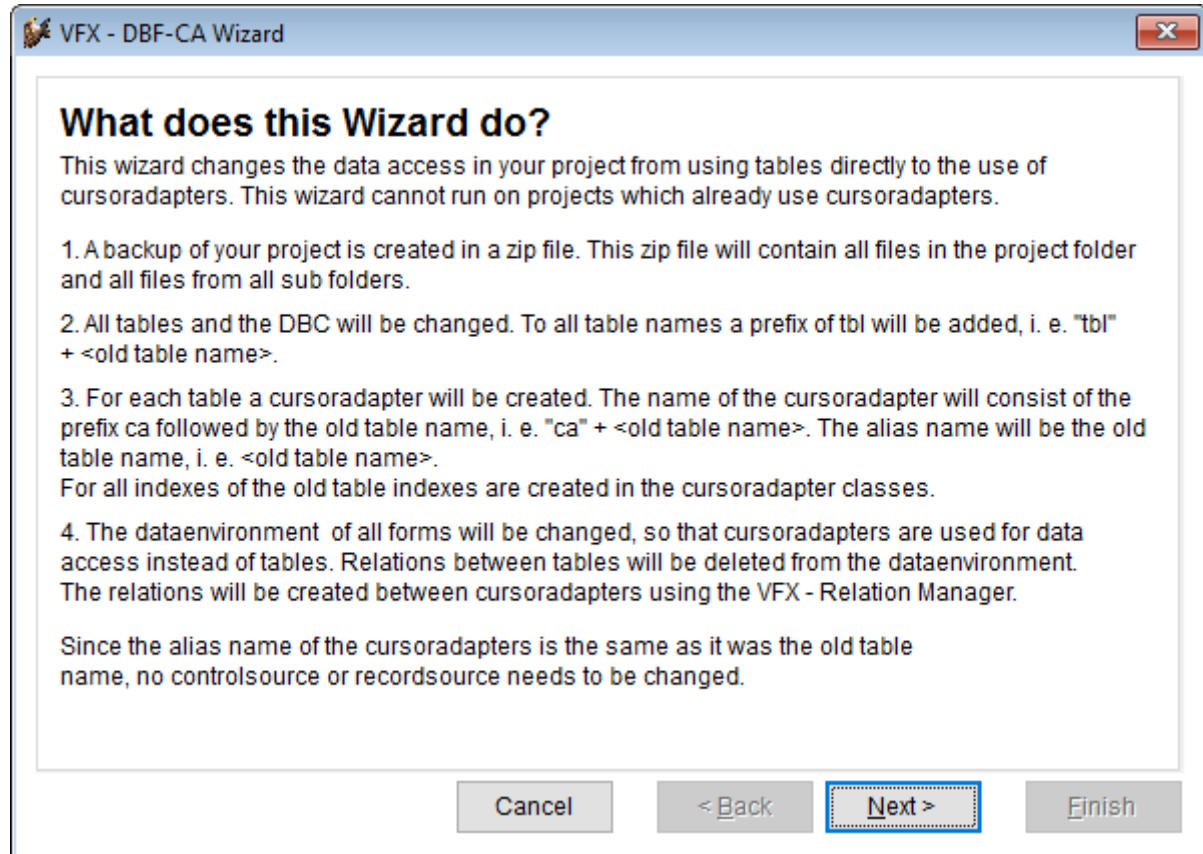
Umstellung ausgehend von Cursoradaptern

Wenn der Datenzugriff ausschließlich mit Cursoradaptern gemacht wird, ist die Umstellung auf eine andere Datenbank am einfachsten.

VFX – DBF-CA Wizard

Der VFX DBF-CA Wizard baut ein ganzes Projekt so um, dass eine Anwendung, die bisher direkt auf VFP Tabellen zugreift, nach dem Lauf des Assistenten mit Cursoradaptern auf die Daten zugreift.

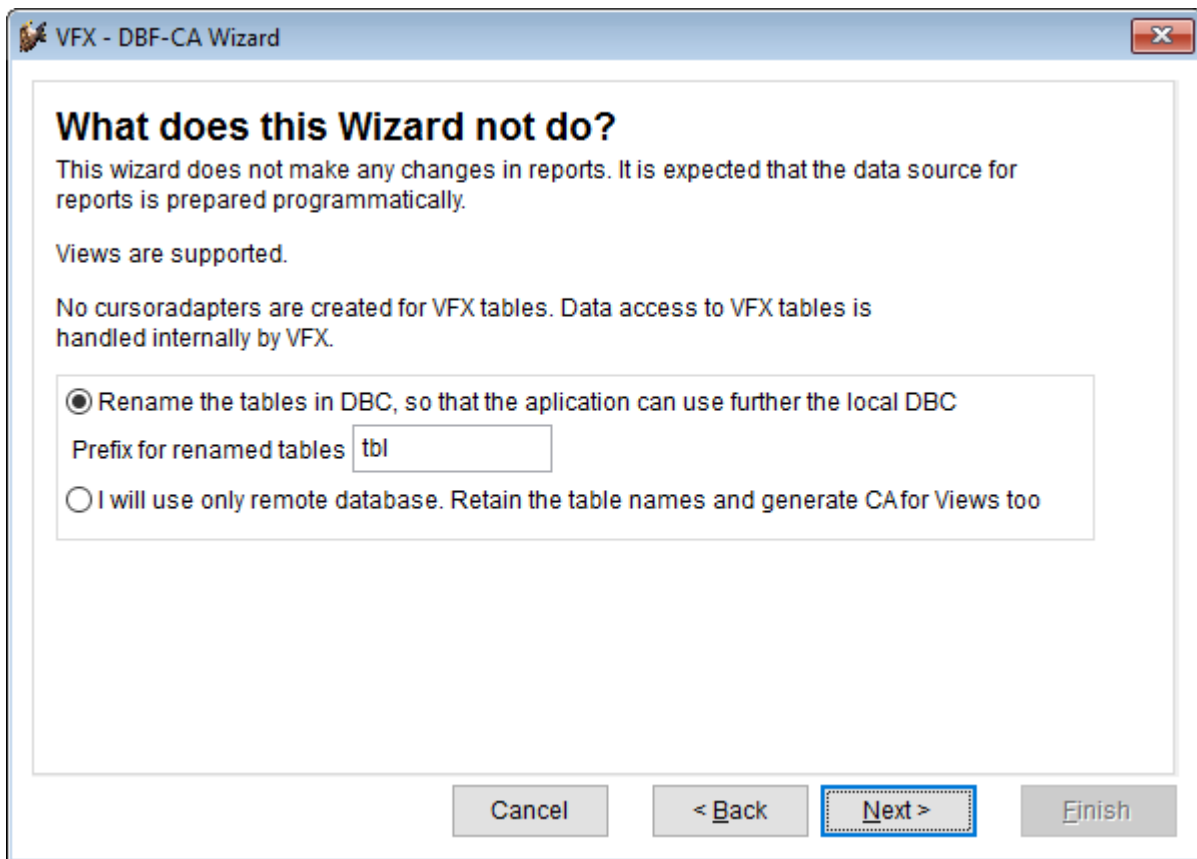
Diese Umstellung bereitet die Anwendung auf den Zugriff auf remote Datenbanken vor.



Im ersten Schritt des Assistenten wird der Anwender über die Funktion des Assistenten informiert.

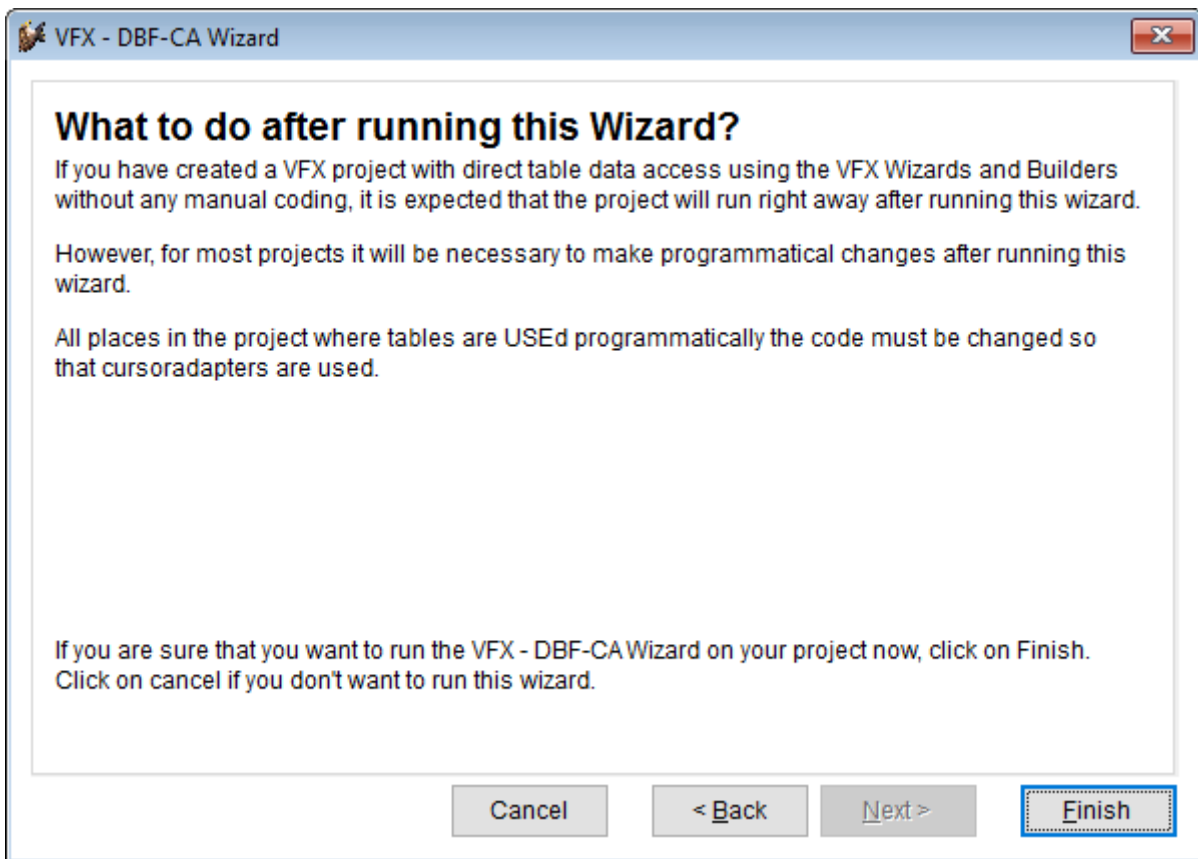
Zunächst wird eine Sicherung des Projektordners und aller Unterordner in eine Archivdatei gemacht. Ggf. kann der Anwender so später wieder auf die Quellen zugreifen, die vor der Ausführung des Assistenten aktuell waren.

Dann wird zunächst die Datenbank umgebaut. Bei der direkten Verwendung von Tabellen befindet sich die Tabelle in der Datenumgebung eines Formulars. Der Aliasname entspricht dem Tabellennamen. Dieser Aliasname wird in den Controlsources der Steuerelemente des Formulars verwendet. Wenn nun mit Cursoradaptern gearbeitet werden soll, muss die verwendete Tabelle zunächst intern vom Cursoradapter geöffnet werden. Hierbei wird der Aliasname der Tabelle verwendet. Der Cursor, den der Cursoradapter bereitstellt, muss daher einen anderen Aliasnamen bekommen. Aus diesem Grund muss der VFX – DBF-CA Wizard die Datenbank umbauen. Allen Tabellennamen wird ein Präfix vorangestellt. Der Standardpräfix ist tbl. Dieser Präfix kann aber auf Wunsch des Entwicklers geändert werden. Die Cursoradapter können dadurch den ursprünglichen Tabellennamen als Aliasnamen verwenden. Alle Steuerelemente können also ihre Controlsource behalten und auch programmatischer Code, der den Aliasnamen verwendet, kann unverändert bleiben.



Wir erwarten, dass unsere Anwendung nach Ausführung des Assistenten weiterhin lauffähig ist. Und das ist sie auch.

Spannend wird es, wenn das Upsizing der VFP Datenbank auf eine SQL Server Datenbank durchgeführt wird und die Anwendung erstmals mit SQL Server Datenbank läuft. Dies wird grundsätzlich möglich sein, allerdings sind beim programmatischen Zugriff auf Daten Änderungen im Code erforderlich.



Beim direkten Zugriff auf eine Tabelle wird diese mit dem Befehl USE geöffnet. Beispiel:

```
USE tabelle
```

Bei der Verwendung eines Cursoradapters, muss zunächst ein Objekt der Cursoradapter-Klasse instanziiert werden. Hierbei erfolgt jedoch noch kein Zugriff auf die Datenbank. Erst durch die Ausführung der Methode `CursorFill()` werden die Daten aus der Datenbank gelesen und es wird ein VFP Cursor erstellt, auf den über den Aliasname zugegriffen werden kann, der in der Eigenschaft `Alias` angegeben ist. In der Regel ist der Aliasname der ursprüngliche Tabellename.

```
loTabelle = NEWOBJECT("catbltabelle")  
  
loTabelle.CursorFill()
```

VFX – Cursoradapter Wizard

Mit dem VFX – Cursoradapter Wizard wird für jede Tabelle einer Datenbank eine Cursoradapterklasse mit Standardeinstellungen erstellt. Die Cursoradapterklassen werden von der VFX Cursoradapterklasse abgeleitet. Dadurch erben alle so erstellten Cursoradapterklassen die VFX Funktionalität für den Datenzugriff. Das heißt insbesondere, dass die Cursoradapter sowohl auf VFP Datenbanken, als auch auf SQL Server Datenbanken zugreifen können.

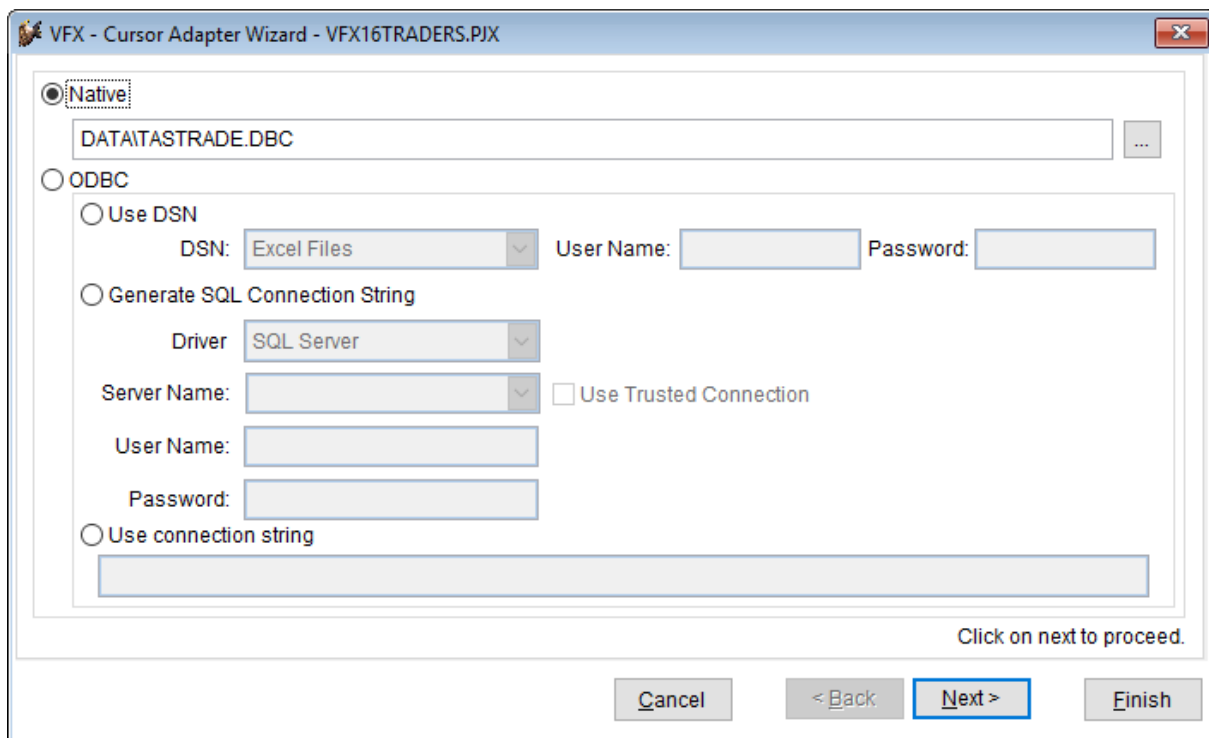
Im ersten Schritt des Assistenten wird die Quelldatenbank ausgewählt. Basierend auf den Tabellen und den Strukturen dieser Tabellen werden die Cursoradapter-Klassen generiert.

Als Quelldatenbank kann eine VFP Datenbank oder eine remote Datenbank dienen.

Wenn bereits eine SQL Server Datenbank vorhanden ist, können also Cursoradapter-Klassen für eine neu zu erstellende Anwendung generiert werden.

Wenn als Quelldatenbank eine SQL Server Datenbank verwendet wird, muss die Verbindung über einen ODBC-Treiber hergestellt werden, dessen Version der Version des SQL Servers entspricht. Für SQL Server 2012 muss der ODBC-Treiber „SQL Server Native Client 11.0“ verwendet werden. Wenn ein älterer Treiber verwendet wird, können nicht alle Funktionen des SQL Servers verwendet werden. Insbesondere können dann nur die Feldtypen verwendet werden, die der ODBC-Treiber unterstützt.

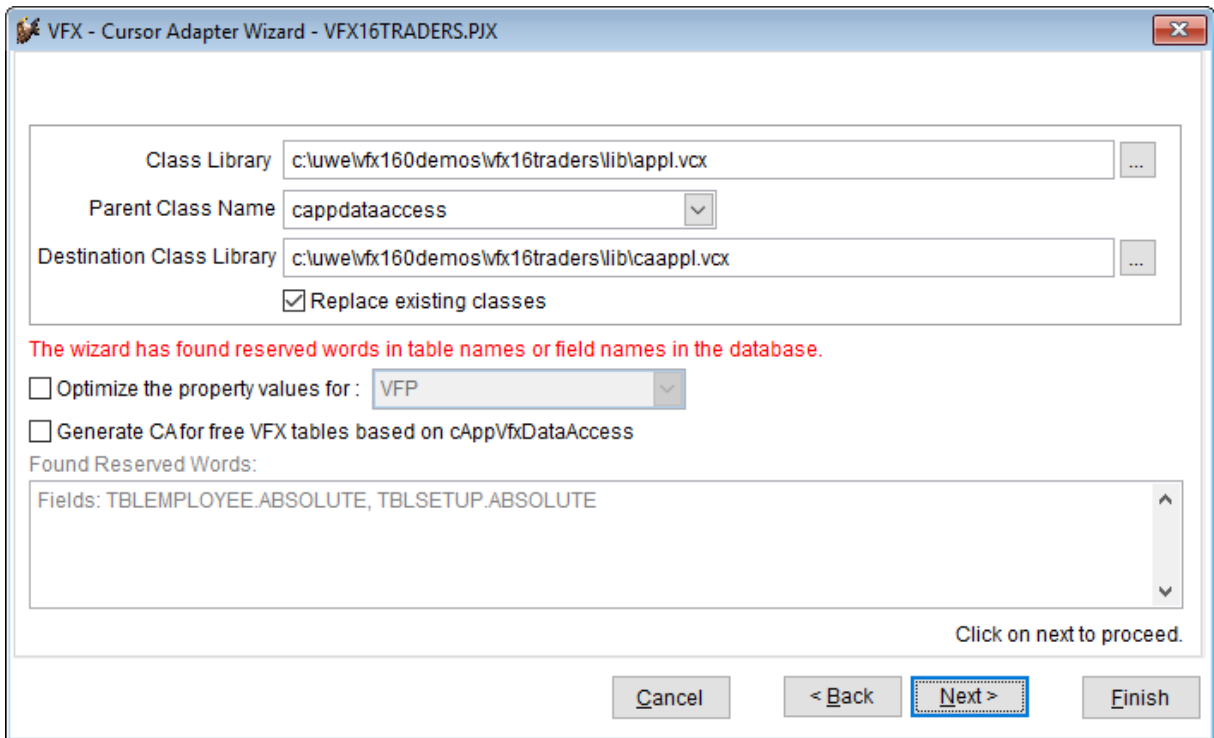
Wenn an der Struktur der Quelldatenbank Änderungen vorgenommen werden, muss der VFX – Cursoradapter Wizard erneut ausgeführt werden, damit die Cursoradapter-Klassen von der neuen Struktur „erfahren“. Änderungen an Formularen und im Programmcode sind nur erforderlich, wenn die geänderten Strukturen verwendet werden sollen.



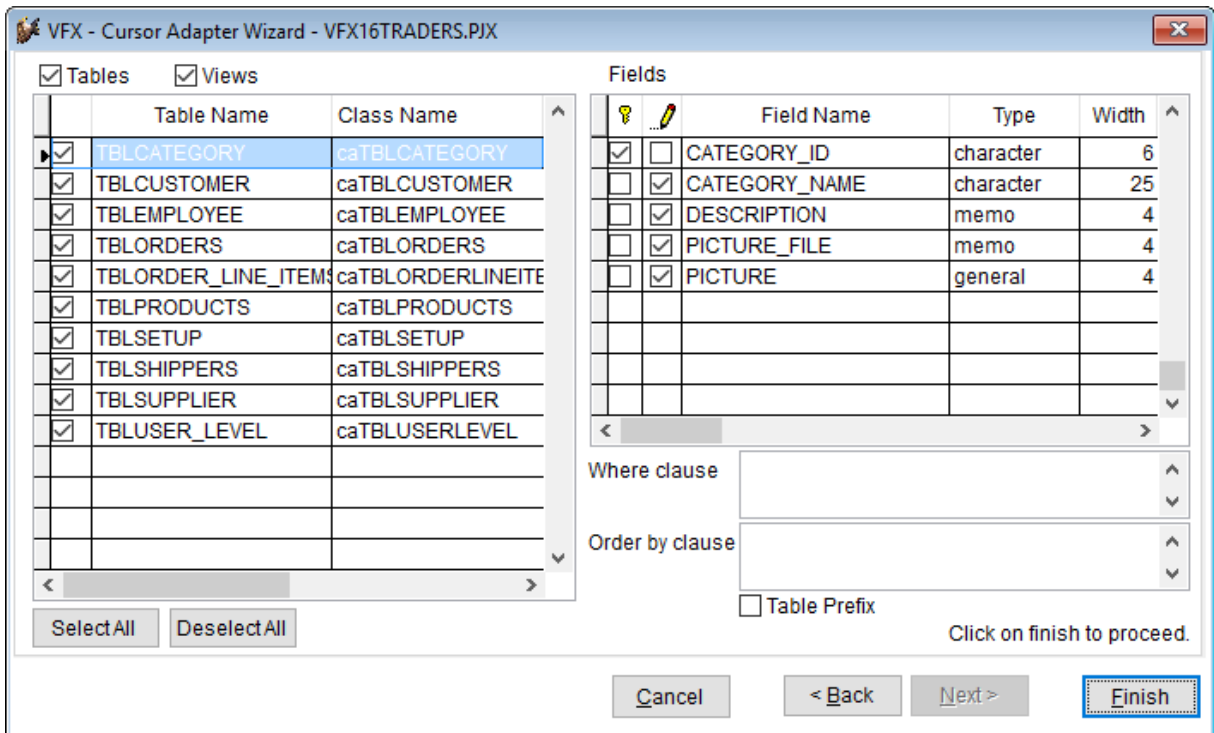
Im zweiten Schritt des Assistenten wird die Klasse ausgewählt, die als Basis für die zu generierenden Cursoradapter verwendet werden soll. Normalerweise ist dies die Klasse `cappdataaccess`, die von VFX zur Verfügung gestellt wird. Wenn ein Entwickler eine eigene Cursoradapter-Klasse verwenden will, kann diese hier ausgewählt werden. Eine eigene Cursoradapter-Klasse sollte jedoch auf der VFX Cursoradapter-Klasse basieren, damit alle erwarteten Eigenschaften vorhanden sind.

Wenn Tabellennamen oder Feldnamen geschützte Wörter sind, werden diese Namen angezeigt. Geschützte Wörter sollten Tabellennamen oder Feldnamen eigentlich nicht verwendet werden. Je nach verwendeter remote Datenbank müssen geschützte Wörter in Abgrenzungszeichen eingeschlossen werden. Beim SQL Server müssen eckige Klammern als Abgrenzungszeichen verwendet werden. Bei der Verwendung von VFP Datenbanken sind keine Abgrenzungszeichen erforderlich.

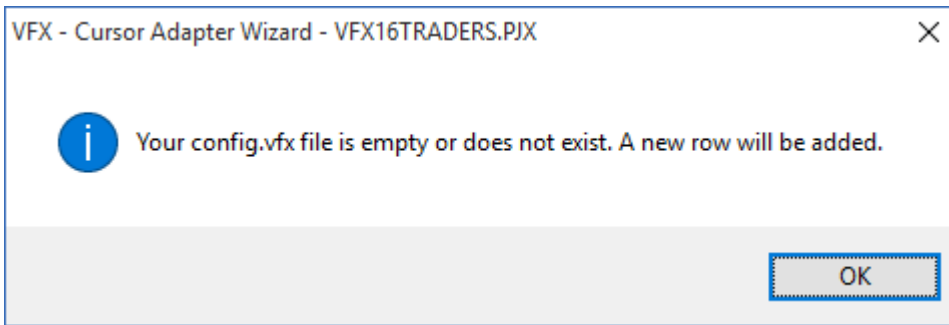
Die bevorzugt oder am häufigsten verwendete Datenbank kann in einer Combobox ausgewählt werden. Für diese Datenbank werden die Abgrenzungszeichen direkt in die Eigenschaften des generierten Cursoradapters geschrieben. Wenn zur Laufzeit eine andere Datenbank verwendet wird, wird Code ausgeführt, der die Eigenschaften mit den benötigten Abgrenzungszeichen neu setzt. Grundsätzlich kann also jede Datenbank verwendet werden, ohne dass es zu Laufzeitfehlern kommt. Für die ausgewählte Datenbank ist der Zugriff jedoch optimiert und es wird zur Laufzeit kein Code ausgeführt, der die Werte der Eigenschaften ändert. Die Ausführung wird geringfügig schneller.



Im dritten Schritt des Assistenten werden die Strukturen der Tabellen der Quelldatenbank angezeigt. Zu jeder Tabelle wird die vorgeschlagene Struktur des zu generierenden Cursoradapters angezeigt. Bei Bedarf kann die Struktur hier geändert werden.



Nach Generierung der Cursoradapter-Klassen legt der VFX – Cursoradapter Wizard eine Datei Config.vfx an oder fügt der vorhandenen eine Zeile hinzu, die den Zugriff auf die ausgewählte Quelldatenbank ermöglicht.



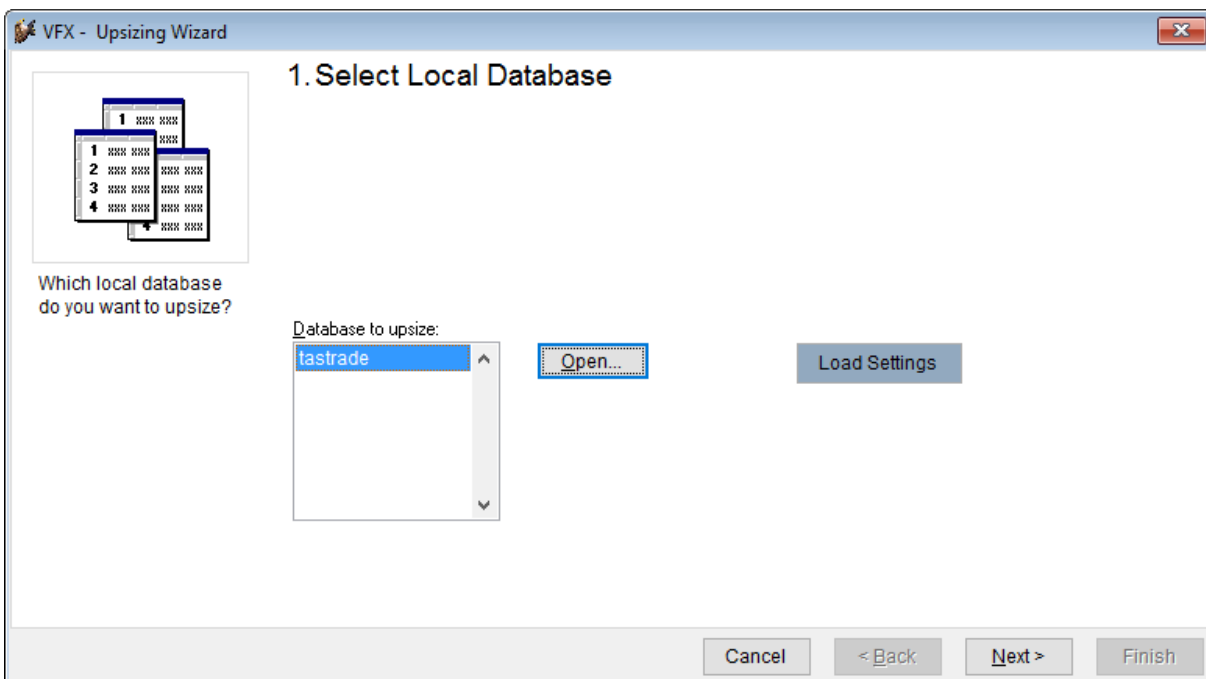
VFX – Upsizing Wizard

Seit der Version 3 wird mit Visual FoxPro ein Upsizing Wizard ausgeliefert. Dieser Wizard ermöglicht es die Struktur und die Daten einer vorhandenen FoxPro Datenbank in eine SQL Server Datenbank zu migrieren.

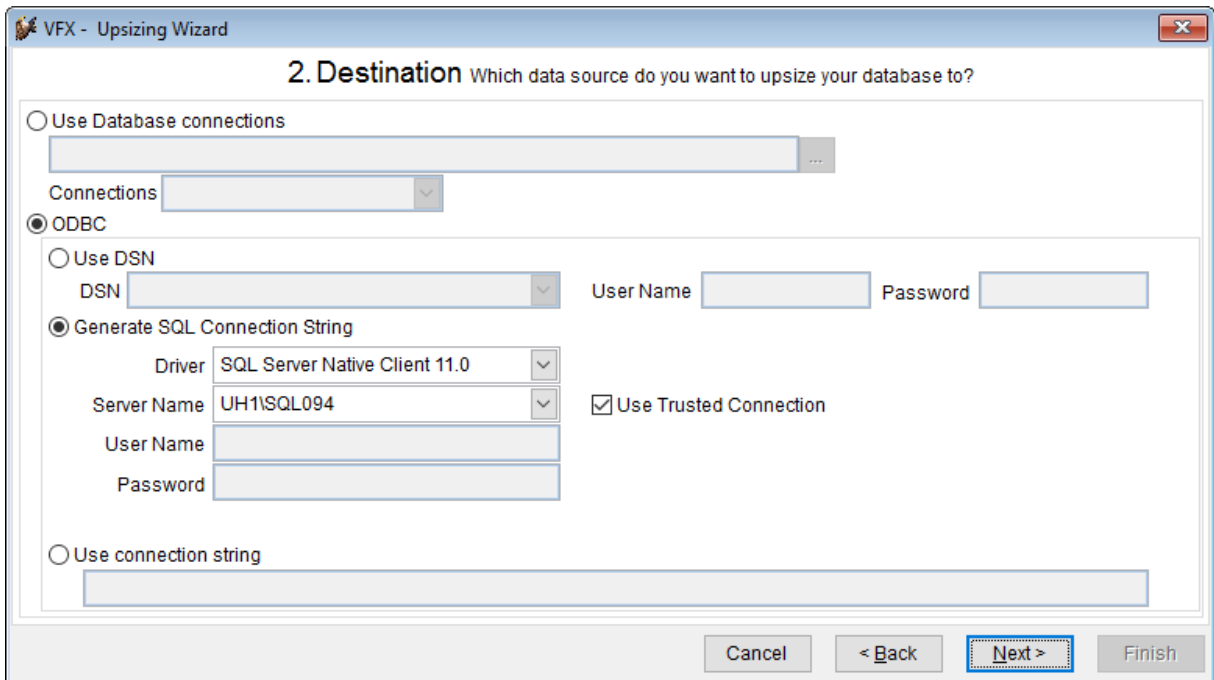
Die erste Version des Upsizing Wizard enthielt einige Fehler, die in der Folgezeit gefixt wurden. Mit neuen Versionen von FoxPro wurden auch neue Versionen des Upsizing Wizard ausgeliefert. Schon immer sind die Quellen des Upsizing Wizard in der Datei Xsource.zip vorhanden, die ebenfalls mit FoxPro ausgeliefert werden. Der geneigte Entwickler kann also den Programmablauf im Quellcode nachvollziehen und bei Bedarf sogar Fehler korrigieren.

Tatsächlich hat sich im Laufe der Jahre gezeigt, dass viele Datenbanken Besonderheiten haben, die im Upsizing Wizard berücksichtigt werden müssen.

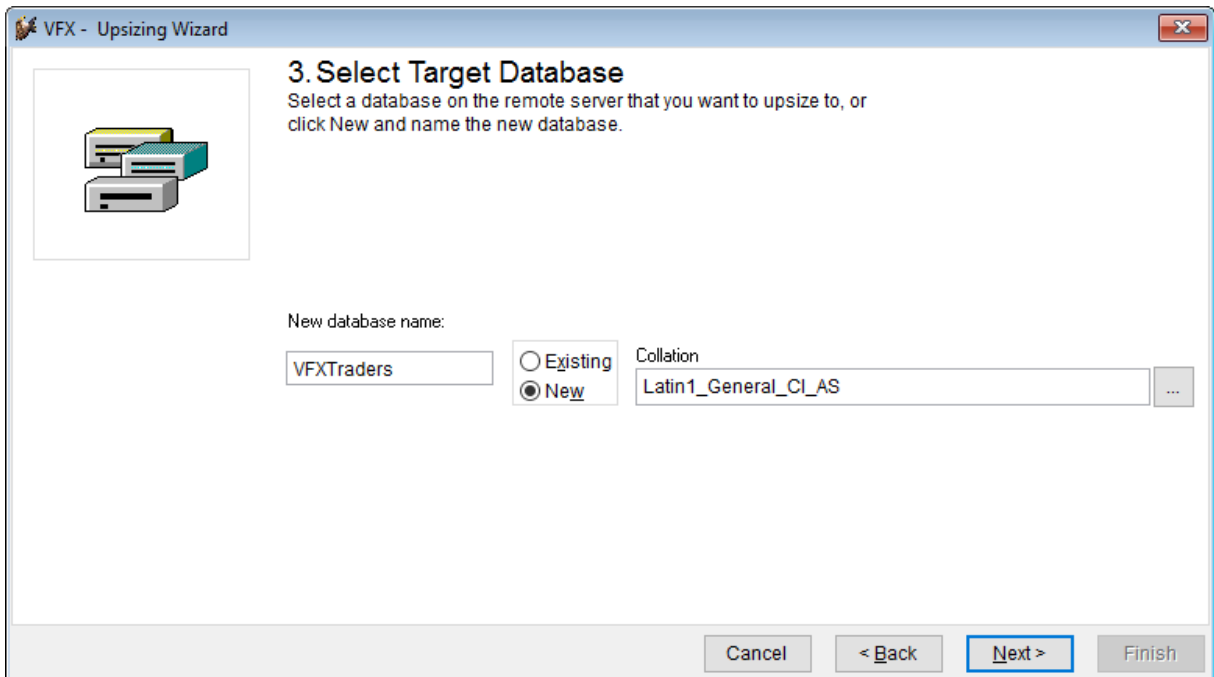
Inzwischen ist der Upsizing Wizard ein Codeplex-Projekt und auf <http://vfx.codeplex.com> zu finden. So sollen Entwickler von aktuellen Verbesserungen profitieren.



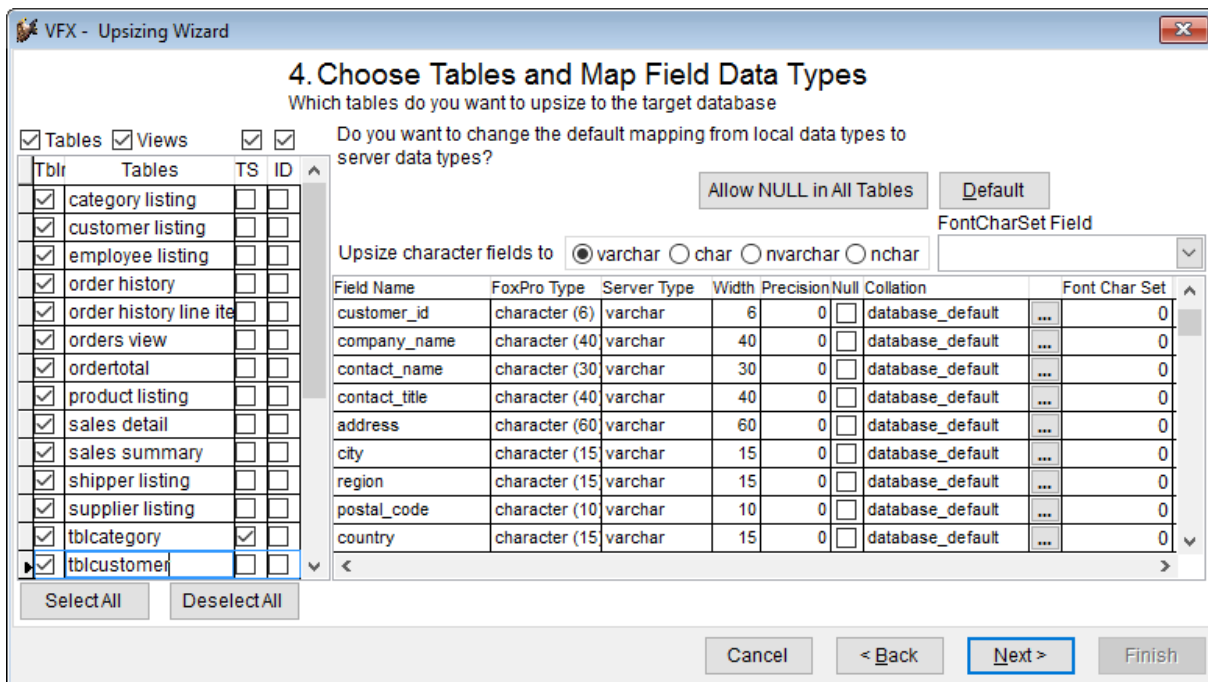
Im ersten Schritt des Upsizing Wizard wird die VFP Quelldatenbank ausgewählt, die nach SQL Server portiert werden soll.



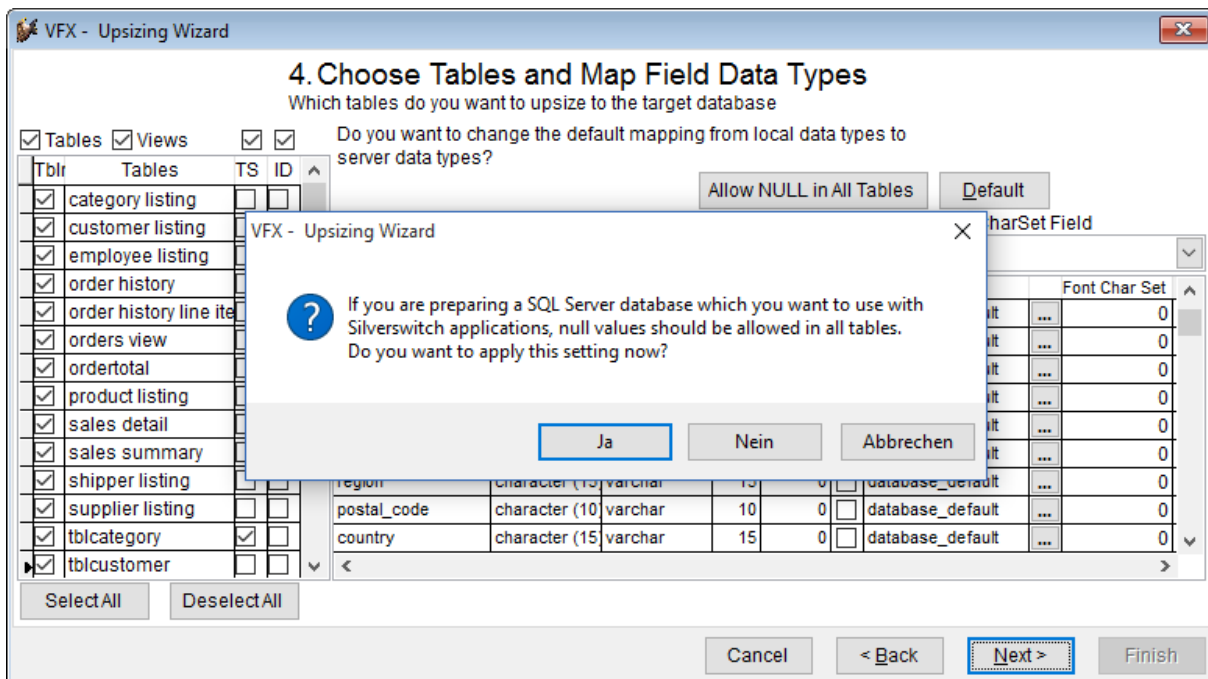
Im zweiten Schritt wird die Verbindung zum SQL Server hergestellt. Diese Verbindung wird ausschließlich vom Upsizing Wizard verwendet und hat nichts mit der späteren Ausführung der Anwendung zu tun.



Anschließend muss der Name der neuen SQL Server Datenbank ausgewählt werden. Im Ausnahmefall kann auch eine vorhandene Datenbank überschrieben werden.

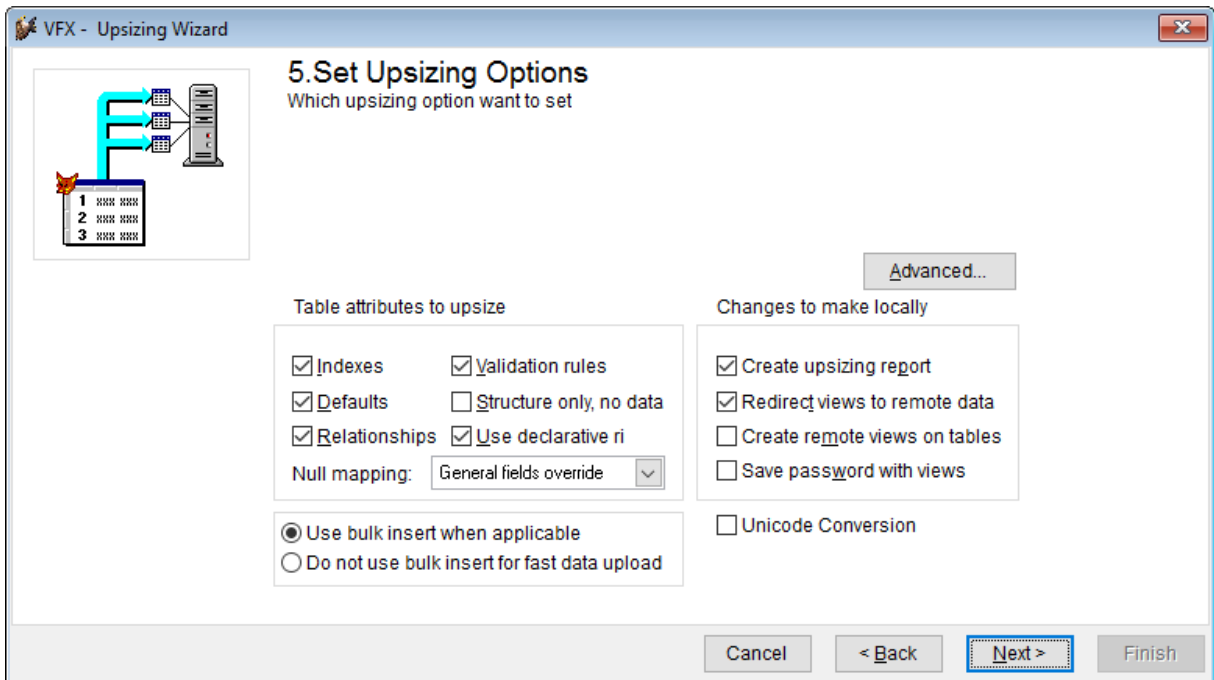


Im vierten Schritt wird die Struktur der Tabellen der Quelldatenbank angezeigt. Zu jedem Feld wird die Struktur des Feldes in der SQL Server Datenbank angezeigt. Bei Bedarf kann die Struktur der Zieldatenbank hier geändert werden.



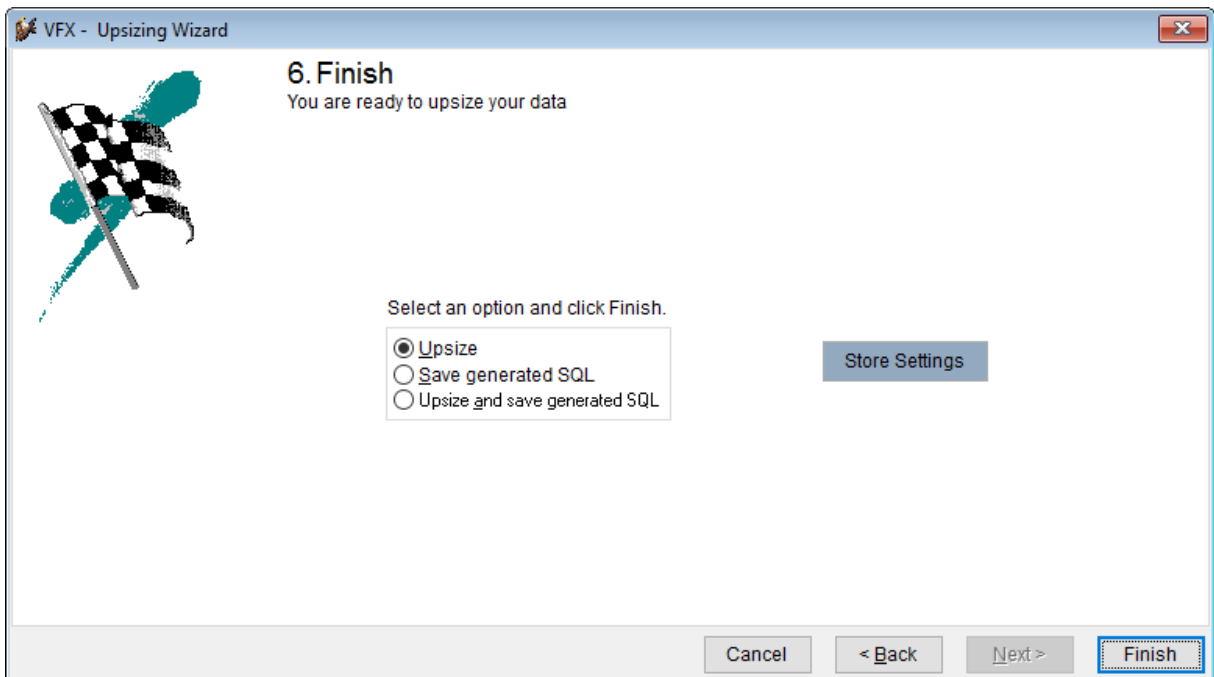
Vor einigen Jahren haben wir begonnen Anwendungen zu entwickeln, die mit anderen Technologien auf die SQL Server Datenbanken zugreifen, die wir mit dem Upsizing Wizard generiert haben. Andere Technologien erwarten in der Regel, dass alle Felder den Status NULL erlauben. Für FoxPro Anwendungen ist dies nicht zwingend erforderlich, aber empfehlenswert.

Wenn in allen Feldern NULL erlaubt ist, können optimierte INSERT und UPDATE Befehle an die SQL Server Datenbank gesendet werden, die nur die Feldinhalte übertragen, die Werte enthalten.



Die Optionen, die im letzten Schritt gesetzt werden können, brauchen normalerweise nicht geändert werden.

Wichtig ist hier die Option „Create upsizing report“, die nach Ausführung des Assistenten Berichte erstellt. Hierbei wird ein VFP Projekt generiert, das verschiedene Berichte enthält, aus denen Informationen über das Upsizing entnommen werden können.



Die meisten Entwickler werden den Assistenten ausführen und erhalten als Ergebnis die gewünschte SQL Server Datenbank. Wahlweise können aber auch nur die generierten SQL Befehle in einer Datei gespeichert werden, die dann zu einem beliebigen Zeitpunkt vom SQL Server Management Studio ausgeführt werden kann.

Nach Beendigung des Upsizing erscheint die Meldung: Upsizing Complete.

Die Ausführungsdauer des Upsizing hängt von im Wesentlichen der Größe der Datenbank ab. Für das Upsizing einer großen Datenbank sollte entsprechend Zeit eingeplant werden. Das Upsizing einer Kundendatenbank sollte in der Regel über ein Wochenende durchgeführt werden.

Nur wenn beim Upsizing Fehler aufgetreten sind, befindet sich der Bericht rpterr1.frx im Projekt. In diesem Bericht werden die Fehler mit den Details angezeigt.

Wenn keine Fehler aufgetreten sind, wird dieser Bericht nicht generiert.

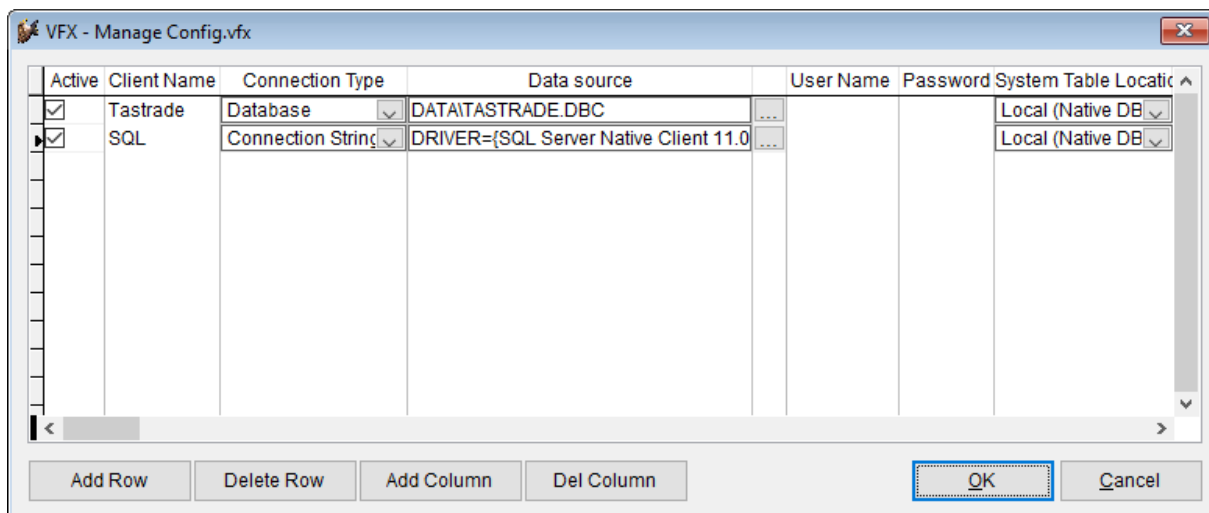
VFX – Manage Config.vfx

Beim Start muss eine Anwendung wissen, mit welcher Datenbank gearbeitet werden soll. Bei VFX Anwendungen ist die Information in einer Konfigurationsdatei gespeichert. Die Informationen in dieser Datei sind in einer XML Zeichenfolge gespeichert und verschlüsselt im Ordner der Anwendung in der Datei Config.vfx abgelegt.

Beim Anwendungsstart wird der Inhalt der Datei gelesen. Wenn sich mehr als ein Datensatz in der Datei befindet, wird dem Anwender ein Dialog angezeigt, in dem er auswählen kann, mit welcher Datenbank gearbeitet werden soll. Die Information über die verwendete Datenbank wird der VFX Cursoradapter-Klasse zur Verfügung gestellt. Somit „weiß“ jeder Cursoradapter, egal ob er sich in der Datenumgebung eines Formulars befindet oder programmatisch instanziiert wird, mit welcher Datenbank er arbeiten soll.

Es sind Einträge möglich, die auf eine VFP Datenbank verweisen. Es können aber auch Einträge angelegt werden, die auf eine SQL Server Datenbank zugreifen lassen.

Die Bearbeitung der Konfigurationsdatei wird mit dem Assistenten VFX – Manage Config.vfx gemacht. Dieser Assistent steht in der Entwicklungsumgebung, aber auch zur Laufzeit in der Anwendung zur Verfügung. Anwender mit Administratorrechten für die Anwendung haben so die Möglichkeit den Datenzugriff zu bearbeiten.



Die Syntax einer Verbindungszeichenfolge kann sich nicht jeder merken. Darum gibt es in VFX einen Generator, der aus interaktiven Eingaben eine Verbindungszeichenfolge generiert.

Driver: SQL Server Native Client 11.0
Server Name: UH1\SQL094
 Use Trusted Connection
User Name:
Password:

Cancel < Back Next > Finish

Nach Auswahl des zu verwendenden SQL Servers und Eingabe der Anmeldedaten kann die Datenbank ausgewählt werden.

Database name VFXTrades

Cancel < Back Next > Finish