



---

*Session D-SP1*

# Microsoft Visual FoxPro 8.0 Service Pack 1 Readme

*Mathias Gronau*

---

## Microsoft Visual FoxPro 8.0 Service Pack 1 Readme

Dieses Service Pack ist ein Update zur ursprünglich erschienen Version von Visual FoxPro 8.0. Es behandelt eine Vielzahl bekannter Probleme mit dem Produkt.

Dieser Artikel enthält die aktuellsten Informationen des Visual FoxPro-Produktteams. Zusätzliche Informationen über Visual FoxPro finden Sie auf der Visual FoxPro-Homepage unter <http://msdn.microsoft.com/vfoxpro>.

### Installation des Visual FoxPro Service Pack 1

**Beachten Sie bitte:** Bevor Sie das Visual FoxPro 8.0 Service Pack 1 installieren können, müssen Sie Visual FoxPro 8.0 installiert haben. Das Service Pack kann nur mit der endgültigen Version von Visual FoxPro 8.0 installiert. Sie können damit keine bestehende Betaversion aktualisieren werden.

#### ***So installieren Sie das Service Pack***

1. Um während der Installation über eine Referenz zu verfügen, drucken Sie diese Datei aus.
2. Auf der Downloadseite des Visual FoxPro 8.0 Service Pack 1 klicken Sie auf die Schaltfläche Download.
3. Akzeptieren Sie die Lizenzvereinbarung. Dieser Schritt ist erforderlich, um das Service Pack herunterladen zu können.
4. Wählen Sie, ob die Datei geöffnet oder auf dem lokalen Rechner gespeichert werden soll.  
Wenn Sie Öffnen wählen, wird Setup automatisch aufgerufen.

Wenn Sie speichern wählen, wechseln Sie in das Verzeichnis, in dem Sie die Datei VFP80SP1.EXE gespeichert haben und klicken doppelt darauf, um das Setup des Service Packs zu starten.

5. Folgen Sie den Anweisungen auf dem Bildschirm, um die Installation zu vollenden.

# In Visual FoxPro 8.0 Service Pack 1 geänderte und hinzugefügte

## Dateien

Die folgenden Dateien wurden für Service Pack 1 aktualisiert:

VFP8.EXE  
VFP8R.DLL  
VFP8T.DLL  
VFPOLEDB.DLL  
RIBUILDR.APP  
REDIST.TXT  
TASKPANE.APP  
VFP8ENU.DLL  
VFP8CHT.DLL  
VFP8CHS.DLL  
VFP8KOR.DLL  
VFP8RENU.DLL  
VFP8RCHS.DLL  
VFP8RCHT.DLL  
VFP8RCSY.DLL  
VFP8RDEU.DLL  
VFP8RESN.DLL  
VFP8RFRA.DLL  
VFP8RKOR.DLL  
VFP8RRUS.DLL  
VFP8Runtime.MSM  
VFPOLEDB.MSM  
VFP8RCHS.MSM  
VFP8RCHT.MSM  
VFP8RCSY.MSM  
VFP8RDEU.MSM  
VFP8RESN.MSM  
VFP8RFRA.MSM  
VFP8RKOR.MSM  
VFP8RRUS.MSM

## Im Service Pack 1 behandelte Probleme

Die folgende Liste enthält die Fehler in Visual FoxPro 8.0, die in Service Pack 1 beseitigt sind:

### COM

- Automation – Die Übergabe eines Array an das ADO RecordSet AddNew als Referenz erzeugt den Fehler „The class file () associated with this field cannot be found - class association cleared“.
- Server – Beim Aufruf einer Methode von einem zweiten Server aus, der in einer MTDLL unter COM+ instantiiert wurde, tritt der Fehler „Unknown Error“ auf.
- Server – Fehler mit COM Codebook in COM+.
- Server – Rückgabe eines nicht typgebundenen Elementarray schlägt fehl.

### Klassen

- Collection – Der Aufruf der Methode AddProperty eines Elements, das einer Collection hinzugefügt wurde, bewirkt, dass die Prozedur eine Objektreferenz zurückgibt.
- Combobox – Die Auswahl funktioniert nicht richtig, wenn die Quelle der Einträge mehr als 512 Elemente umfasst.
- Combobox – Ein Wert mit einem Et-Zeichen (&) fügt dem angezeigten Wert ein zweites Et-Zeichen hinzu.

- Editbox – C5 fatal exception error tritt auf, wenn eine Editbox mit einem Feld vom Typ Zeichen als Datenquelle aktualisiert wird.
- Form – Zur Design- und zur Laufzeit tritt ein Fehler auf, wenn der Wert einer Eigenschaft die Methode eines Objekts referenziert, das sich nicht im Gültigkeitsbereich befindet.
- Form – F5 verfolgt das Ereignis KeyPress des Formulars nicht, wenn die Eigenschaft Closable des Formulars auf .F. gesetzt ist.
- Form – Das Menü Steuerelemente enthält unter Windows NT 4.0 falsche Hotkeys.
- Grid – Fatal Exception Error tritt auf, wenn eine Spalte in Containerobjekt enthält.
- Grid – Die Spalten werden zur Design- wie zur Laufzeit nicht korrekt angezeigt, wenn die Eigenschaft RightToLeft auf .T. gesetzt ist.
- Grid – Die Eigenschaft HighlightForeColor erscheint weiß, wenn sie auf schwarz gesetzt ist.
- Grid – Ist die Eigenschaft AllowCellSelection auf .F. gesetzt, deaktiviert sie die Eigenschaft Cancel der Befehlsschaltfläche.
- Grid – Mit einem CursorAdapter-Objekt wird zweimal ein Fehler angezeigt und das Formular schließt unerwartet.
- Grid - Fatal Exception Error tritt auf, wenn ein Grid mit einigen Textspalten geöffnet wird und Active Accessibility aktiviert ist (OBJ\_SHOW).
- Grid – wird nicht aktualisiert, wenn er sich innerhalb eines Pageframes auf einer nicht aktiven Seite befindet.
- Grid – Das Ereignis MouseUp verhält sich nicht konsistent, wenn auf eine Trennlinie im Header geklickt oder doppelgeklickt wird und die Eigenschaft AllowAutoColumnFit gesetzt ist.
- Header – Eine programmgesteuert definierte Headerklasse in einer .EXE-Datei wird nicht gefunden, es sei denn, die .PRG-Datei, die die Definition enthält, befindet sich im gleichen Verzeichnis wie die .EXE.
- Label – Das Setzen der Eigenschaft Caption auf eine vorgegebene Eigenschaft mit einem leeren Wert, also beispielsweise .Caption='thisform.tag', erzeugt den Fehler „There is not enough memory to complete this operation.“
- Label – Wird die Größe eines Labels verändert, tritt der Fehler “There is not enough memory to complete this operation” auf und Visual FoxPro verschwindet.
- Listbox – Ist die Eigenschaft RowSourceType auf 8 gesetzt, tritt ein Speicherleck auf.
- Pageframe – Fatal Exception Error tritt auf, wenn ein Formular geschlossen wird, nachdem auf eine Seite geklickt wurde, in der die Eigenschaft PageOrder gesetzt ist.
- Pageframe – Ist die Eigenschaft TabOrientation eines Pageframes auf 2 gesetzt, bewirkt das Benutzen der Pfeiltasten für das Wechseln zwischen den Tabs auf einem Formular mit Themes ein Speicherleck und Visual FoxPro stürzt unweigerlich ab.
- Textbox – Der Fehler „Not a character expression“ tritt auf, wenn der Wert einer Textbox geändert wird, in der ein Element eines Arrays als Datenquelle gesetzt ist.
- XMLAdapter – Ignoriert die Eigenschaft XMLNamespace, es sei denn, XMLName ist nicht leer und schlägt beim Parsen des XML fehl.
- Verschiedene – Die Methode setzt den Wert der bestehenden Eigenschaft auf .F., wenn kein neuer Wert übergeben wurde.
- Verschiedene – Der Fehler „Variable not found“ oder „data type is invalid for this property“ tritt auf, wenn die Eigenschaft Caption eine Variable referenziert, die sich nicht im Wirkungsbereich befindet.
- Verschiedene – Wird ein Leerstring als Caption angegeben, wird keine leere Caption angezeigt, sondern der Objektname.
- Verschiedene - Fatal Exception Error tritt auf, wenn ein Objektarray auf mehr als 32.767 Elemente dimensioniert wird.
- Verschiedene - Fatal Exception Error tritt auf, wenn die benutzerdefinierten Eigenschaften FontBold und FontItalic hinzugefügt werden und diese dann im Eigenschaften-Fenster gesetzt werden.

- Verschiedene - Fatal Exception Error tritt auf, wenn in der Methode Destroy eine freigegebene Variable erneut definiert wird.
- Verschiedene - Fatal Exception (buffer overrun) Error tritt auf, wenn eine abgeleitete Methode mit einer Parameterliste geändert wird, die mehr als 256 Zeichen lang ist.

## Daten/Engine

- Auto Increment – Funktioniert mit APPEND FROM nicht richtig.
- CursorAdapter – Fügen Sie Funktionalität hinzu, um Updatekonflikte im CursorAdapter zu behandeln.
- CursorAdapter – Unterstützt der ODBC-Treiber oder OLE DB Provider die Stapelverarbeitung nicht, schlägt auch ein einzelnes Update fehl, wenn die Eigenschaft UpdateType auf 2 gesetzt ist.
- CursorAdapter – Schlägt beim Update einer .MDB mit Länge Null, ohne Felder oder mit mehr als 255 Memofeldern in ADO fehl.
- CursorAdapter – Schlägt in ADO die Aktualisierung einer Tabelle in einer Funktion TABLEUPDATE() fehl, führt dies dazu, dass auch die weiteren Aufrufe von TABLEUPDATE() fehlschlagen.
- CursorAdapter – Wenn der OLE DB Provider beim Füllen der Collection Parameters fehlschlägt, ist er nicht in der Lage, dem Objekt ADODB.Command einen Leerstring als Parameter zu übergeben.
- CursorAdapter – Es ist möglich, die Funktion TABLEREVERT() aufzurufen, obwohl TABLEUPDATE() ausgeführt wird.
- CursorAdapter – Die Funktion TABLEUPDATE() übergibt nicht an lokale Cursor, wenn Buffering auf 5 gesetzt und der Datensatz im Ereignis AfterInsert geändert wurde.
- CursorAdapter – Der Prozess des Beziehens von Daten durch ADO versetzt den Tabellenpuffer in einen nicht korrekten Zustand.
- Engine – Der Fehler „Index does not match the table“ tritt bei einem GO TOP in einem tabellengepufferten Cursor auf, nachdem BEGIN TRANS und ROLLBACK aufgerufen wurden.
- Engine – Der Fehler „Record is not in index“ tritt während eines SCAN oder während des Springens durch die Datensätze auf.
- Engine – Der Fehler „Table has no index order set“ tritt bei einem geöffneten tabellengepufferten Cursor auf, der einen Datensatz enthält, dessen Felder geändert wurden.
- Engine – Der Befehl SEEK mit der Klausel ORDER DESCENDING bewegt den Datensatzzeiger nicht zum richtigen Datensatz, wenn SET NEAR auf ON steht.
- Engine – Der Befehl SEEK findet alte und neue Daten mit optimistischer Tabellenpufferung.
- Engine – Der Fehler „Index does not match the table...(Error 114)“ tritt auf, wenn die Funktion TABLEUPDATE() auf einem tabellengepufferten Cursor mit gepufferten Änderungen aufgerufen und der Befehl REINDEX eingesetzt wird.
- Engine – Ein Speicherleck tritt auf, wenn mit der Funktion SQLEXEC() Parameter als Referenzen übergeben werden.
- Engine – Ein Speicherleck tritt auf, nachdem SQL INSERT oder SQL UPDATE einen Fehler hervorgerufen hat.
- Engine – Die Funktion SQLGETPROP() enthält ein Speicherleck.
- Engine – Wird in der Mitte einer Transaktion ein SQL INSERT aufgerufen, tritt ein schwerer Ausnahmefehler auf.
- Engine – Der Aufruf der Funktion SQLSTRINGCONNECT() ohne Parameter ruft einen Fehler hervor, statt wie in Visual FoxPro 7.0 einen Dialog anzuzeigen.
- Engine – Im SQL Pass-Through Asynchronous Batch-Modus werden alle Ergebnissätze auf einmal abgerufen.
- Engine – Der Befehl COPY TO mit mehr als 255 Feldern ruft einen schweren Ausnahmefehler oder einen Berichtsfehler hervor.

- Engine – Die Funktion REQUERY() auf einer parametrisierten Ansicht schlägt fehl, wenn sie mit einem Gleichheitszeichen (=) begonnen wird.
- Engine – Nach dem Aufruf von TABLEUPDATE() und ROLLBACK in einer lokalen Ansicht tritt beim Befehl GOTO TOP der Fehler „Index does not match the table“ auf.
- ODBC – Beim Versuch, im SQL Server eine ungültige Eigenschaft einzustellen, erscheint eine falsche Fehlermeldung.
- ODBC – Die Verbindung wird nicht unterbrochen, wenn der Leerlauf-Timeout aufgetreten ist.
- ODBC – Die Übergabe einer großen Variablen an den SQL Server ODBC-Treiber kann einen schweren Ausnahmefehler hervorrufen.
- ODBC – Verbindung für den gemeinsamen Zugriff schlägt bei der Übergabe eines Parameters oder im asynchronen Modus fehl.
- SET TABLEVALIDATE – SET TABLEVALIDATE TO setzt TABLEVALIDATE auf einen falschen Wert.
- SET TABLEVALIDATE – Hinzufügen eines neuen Bit, um den Fehler „File is in use“ zu unterdrücken, wenn nLevel auf 1 gesetzt ist.
- SET TABLEVALIDATE – Es tritt kein Fehler auf, wenn ein Datensatz hinzugefügt wird, SET TABLEVALIDATE auf 2 gesetzt ist und der Datensatzzähler niedriger ist als er sein sollte. Der neue Datensatz überschreibt bestehende Daten.
- SQL – UPDATE und DELETE sind nicht Rushmore-optimiert, wenn die Zieltabelle gepufferte Änderungen enthält.
- SQL – Der Fehler „Fields are incompatible“ tritt auf, wenn SELECT...UNION auf eine Fox 2.x-Tabelle ausgeführt wird.
- SQL – Die JOIN-Syntax liefert manchmal falsche Ergebnisse.
- SQL – INSERT INTO ... SELECT ... FROM ... WHERE ... beachtet die Klausel WHERE nicht.
- SQL – CREATE/REMOVE TABLE im DBC schlägt in einer Mehrbenutzerumgebung fehl.
- SQL – Änderungen in Fehlermeldung und –nummer mit nicht ausbalancierten INSERT INTO ... SELECT FROM-Anweisungen.

## IDE

- \_SCREEN – Es tritt ein schwerer Ausnahmefehler auf, wenn die IDE minimiert und wiederhergestellt wird, nachdem dem Ereignis \_SCREEN.Resize ein Hook zugeordnet und wieder entfernt wurde.
- Debugger – Ein schwerer Ausnahmefehler tritt auf, nachdem Werte im Fenster Locals für die Eigenschaft eines geschlossenen modalen Formulars geändert wurden.
- Debugger – Ist der Aufrufstack oder das Fenster Locals nicht geöffnet, tritt ein schwerer Ausnahmefehler auf, nachdem Sie in das Fenster Trace zurückgekehrt sind.
- Debugger – Ein schwerer Ausnahmefehler tritt auf, nachdem die Programmausführung wieder aufgenommen wurde mit einem großen Array im Watch-Fenster.
- Debugger – Ein schwerer Ausnahmefehler tritt auf beim Kopieren eines Ausdrucks in das Watch-Fenster.
- Editor – Ein kritischer Ausnahmefehler tritt auf, wenn während der Änderung einer gespeicherten Prozedur im Kontextmenü wird der Eintrag „Definition anzeigen“ oder „Verweis nachschlagen“ gewählt wird.
- Editor – Ein kritischer Ausnahmefehler tritt auf, wenn Code aus dem Kontextmenü heraus auskommentiert wird.
- Menüs – Ein schwerer Ausnahmefehler tritt auf, wenn nach der Änderung eines Berichts einer der Befehle PUSH MENU, POP MENU oder SET SYSMENU TO DEFAULT aufgerufen wird.
- Menüs – Ein schwerer Ausnahmefehler tritt auf, wenn nach dem Schließen des Debuggers SET SYSMENU TO DEFAULT aufgerufen wird.

- Menüs – Ein schwerer Ausnahmefehler tritt auf, wenn SET SYSMENU SAVE aufgerufen wird, während der Berichts-Designer geöffnet ist.
- Dialog Optionen – Suchpfadeinstellungen, die im Tab Dateiablage im Dialog Optionen vorgenommen und als Vorgabewert gespeichert wurden, bleiben für spätere Visual FoxPro-Sitzungen nicht erhalten.
- Berichts-Designer – Die Schaltfläche Schließen erscheint nicht unter Windows NT 4.0, wenn das Vorschaufenster maximiert ist.
- Assistent für referenzielle Integrität – Generiert einen Fehler „Missing expression“.
- Task Pane – Ein schwerer Ausnahmefehler tritt auf, wenn sich die Seite Community während eines Startprogramms nicht mit einem Webdienst verbinden kann.
- Task Pane – Das Fenster Community kann sich nicht verbinden, wenn IE konfiguriert ist, einen Proxy zu nutzen.
- Toolbars – Schwarze und weiße Toolbars können zu einem schweren Ausnahmefehler führen.

## **OLE DB Provider**

- Die Befehlsausführung ist zwischen allen Prozessen serialisiert.
- IDBCreateSessionImpl::CreateSession prüft nicht den Rückgabewert von IID\_IObjectWithSite und bewirkt einen schweren Ausnahmefehler.
- InterlockedCompareExchange und InterlockedExchangeAdd nutzen nicht den Multithreading-Schutz.
- CFGInitFailed() in Visual FoxPro OLE DB Provider (VFPOLEDB.dll) zeigt die MessageBox.
- Verbesserungen der Threads für den Visual FoxPro OLE DB Provider.
- Stellen Sie die Leseinstellungen des OLE DB Providers für ENGINEBEHAVIOR und TABLEVALIDATE in der CONFIG.FPW ein.
- Speicherleck in VFPOLEDB.dll.
- SET ENGINEBEHAVIOR betrifft alle Verbindungen des gleichen Clients.

## **Anderes**

- Es tritt ein schwerer Ausnahmefehler auf, wenn der Programmcache gefüllt ist.
- Wenn etwas abstürzt, kann der Dialog Fatal Exception mehrfach angezeigt werden.
- Bei SET NOTIFY CURSOR OFF verhält sich \_TALLY so, als wäre es zusätzlich.
- Enthält ein InstallShield Express Setup Mischmodule (MSM) sowohl von Visual FoxPro 7.0 als auch von 8.0, erscheint beim Erstellen des Setups eine Fehlermeldung.
- Der Fehler „Array dimensions are invalid“ tritt auf, wenn in einem Array auf eine Zeile hinter der 4906ten zugegriffen wird.

## **Verbesserungen in Service Pack 1**

Das Visual FoxPro 8.0 Service Pack 1 enthält in folgenden Bereichen Verbesserungen:

- Neue Eigenschaften für die bessere Behandlung von Updatekonflikten in der Klasse CursorAdapter.
- Zusätzliche Optionen für den Befehl SET TABLEVALIDATE.

## **Neue Funktionalitäten in CursorAdapter**

Visual FoxPro 8.0 Service Pack 1 enthält zusätzliche Eigenschaften für die Klasse CursorAdapter, um Updatekonflikte besser behandeln zu können. Dazu gehören die Eigenschaften ConflictCheckType und ConflictCheckCmd.

## Die Eigenschaft ConflictCheckType

Die Eigenschaft ConflictCheckType gibt an, ob die Prüfungen auf Konflikte während einer Update- oder Löschoption durchgeführt werden. Steht zur Design- wie zur Laufzeit zum Lesen und Schreiben zu Verfügung.

```
CursorAdapter.ConflictCheckType [ = nValue]
```

### Parameter

*nValue*

Legt einen Wert fest, mit dessen Hilfe festgestellt wird, die die Konfliktprüfungen durchgeführt werden. Die folgende Tabelle enthält die Werte für *nValue*.

| nValue | Beschreibung  |
|--------|---|
| 0      | Keine Prüfung (Vorgabewert)   |
| 1      | Im Modus, in dem einzelne Datensätze aktualisiert werden, wird die Prüfung auf Updatekonflikte während einer SQL UPDATE oder DELETE-Operation durchgeführt. Tritt ein Konflikt auf, besonders, wenn weniger als ein Datensatz von den Befehlen betroffen ist, die in den Eigenschaften UpdateCmd und DeleteCmd angegeben sind, wird der Fehler „Update conflict (Error 1585)“ zurückgegeben.                              |
| 2      | Im Modus, in dem einzelne Datensätze aktualisiert werden, wird während einer SQL UPDATE oder DELETE-Operation auf die Eindeutigkeit des Schlüssels geprüft. Ist mehr als ein Datensatz von den Befehlen betroffen ist, die in den Eigenschaften UpdateCmd und DeleteCmd angegeben sind, geben Sie die Nachricht „Warning: The key defined by the KeyField property for table "alias" is not unique. (Error 1495)“ zurück. |
| 3      | Führt die in Einstellung 1 und 2 dargestellten Prüfungen aus.   |
| 4      | Hängen Sie den in der Eigenschaft ConflictCheckCmd angegebenen benutzerdefinierten Befehl an die Befehle in den Eigenschaften UpdateCmd und DeleteCmd an.   |

### Bemerkungen

- ConflictCheckType und ConflictCheckCmd werden für native, ODBC- und ADO-Datenquellen eingesetzt. Beim Einsatz von ADO sollte der CursorAdapter für die SQL UPDATE- und DELETE-Operation ein ADO Befehlsobjekt einsetzen. Andernfalls beachtet Visual FoxPro ConflictCheckType und ConflictCheckCmd nicht.
- Beim Ausführen einer TABLEUPDATE()-Operation im Stapel beachtet Visual FoxPro die Einstellungen 1, 2 und 3 für ConflictCheckType nicht.
- Es ist empfehlenswert, die Einstellungen 1, 2 und 3 nicht zu verwenden, wenn die Eigenschaften UpdateCmd und DeleteCmd mehr als einen Befehl enthalten. In diesem Szenario ist das Verhalten des CursorAdapters vom ODBC-Treiber oder OLE DB Provider abhängig und kann zu unerwarteten Ergebnissen führen.
- Die Einstellungen 1, 2 und 3 für ConflictCheckType könnten fehlschlagen, wenn der ODBC-Treiber oder OLE DB Provider nicht zur Verfügung steht, um die erforderliche Funktionalität bereitzustellen, oder die Funktionalität ist durch den Treiber, Provider oder die Backend-Konfiguration deaktiviert. Beispielsweise deaktiviert SET NOCOUNT ON im SQL Server die Funktionalität und verhindert, dass CursorAdapter Aktualisierungen vornimmt, wenn ConflictCheckType auf 1, 2 oder 3 steht.

## Die Eigenschaft ConflictCheckCmd

Die Eigenschaft ConflictCheckCmd enthält eine Referenz auf einen benutzerdefinierten Befehl, wenn die Eigenschaft ConflictCheckType auf 4 steht. Sie kann zur Design- wie zur Laufzeit lesen und geschrieben werden.

```
CursorAdapter.ConflictCheckCmd [ = cValue]
```

## Parameter

### *cValue*

Legt die Referenz auf einen benutzerdefinierten Befehlsstring fest, der an die Befehle angehängt wird, die in den Eigenschaften UpdateCmd und DeleteCmd für die Prüfung von Aktualisierungs- oder Löschkonflikten festgelegt sind. Vorgabewert für *cValue* ist ein Leerstring ("").

## Bemerkungen

Vergleichen Sie dazu den Abschnitt Bemerkungen von ConflictCheckType.

## Beispiel

```
SQL Server: ConflictCheckCmd="IF @@ERROR=0 AND @@ROWCOUNT!=1 RAISERROR (' Update conflict or non-unique key.', 16, 1)"
```

```
Native: ConflictCheckCmd="CheckConflict()"
```

```
FUNCTION CheckConflict()  
    IF _tally<1  
        ERROR(1585)  
    ELSE  
        IF _tally>1  
            ERROR(1495)  
        ENDIF  
    ENDIF  
ENDFUNC
```

## Verbesserte Funktionalität von SET TABLEVALIDATE

Visual FoxPro 8.0 Service Pack 1 enthält zusätzliche Optionen für den Befehl SET TABLEVALIDATE, mit deren Hilfe Sie die Überprüfung der Tabelle während verschiedener Datenoperationen besser steuern können. Aktualisierungen an diesem Befehl, die im Service Pack 1 neu sind, sind in rot dargestellt.

```
SET TABLEVALIDATE TO [nLevel]
```

## Parameter

### *nLevel*

Legt die Stufe der Tabellenprüfung als Summe der angegebenen Bitwerte fest, die in der folgenden Tabelle beschrieben werden. Wenn Sie nLevel nicht angeben, wird die Tabellenprüfung auf die Stufe 3 (Vorgabewert) gesetzt.

| Bit | Bitwerte (additiv) | Beschreibung   |
|-----|--------------------|--|
| 0   | 1                  | Wenn eine Tabelle geöffnet ist, wird der Datensatzzähler geprüft.<br><b>Beachten Sie bitte:</b> Der Header der Tabelle (.dbf) ist während der Prüfung gesperrt. Dies kann zu einer verringerten Skalierbarkeit führen, wenn Tabellen für den gemeinsamen Zugriff geöffnet sind, beispielsweise durch den Aufruf von USE <i>TableName</i> SHARED, oder wenn für eine nicht geöffnete Tabelle der Befehl SQL SELECT, UPDATE, INSERT oder DELETE eingesetzt wird und SET EXCLUSIVE auf OFF steht.   |
| 1   | 2                  | Der Datensatzzähler wird geprüft, wenn Datensätze angehängt (oder eingefügt) und auf die Festplatte geschrieben werden.  |
| 2   | 4                  | Unterdrücken des Fehlers „File is in use (Error 3)“, wenn der Header für eine Operation zum Öffnen nicht gesperrt werden kann. Erfordert, dass das erste Bit gesetzt ist (Ausführen der Prüfung des Datensatzzählers, wenn die Tabelle geöffnet ist).<br>Der Befehl USE, der die Tabelle öffnet, versucht nur ein Mal, den Header zu sperren. Ist er bereits gesperrt, führt Visual FoxPro eine strikte Prüfung aus entsprechend der Einstellung des Befehls SET TABLEVALIDATE auf 1. Andernfalls unterdrückt Visual FoxPro die Anzeige des Fehlers „File is in use“ und führt eine Headerprüfung im Stil von Visual FoxPro 7.0 aus. |
| 3   | 8                  | Der Header der Tabelle (.dbf) wird geprüft, bevor die Einfügeoperation auf der Festplatte gespeichert und der Header geändert wird. Dies betrifft Einfügeoperationen.  |

## Bemerkungen

- Steht SET TABLEVALIDATE auf 0, wird keine Tabellenprüfung ausgeführt.
- SET TABLEVALIDATE enthält keine Funktionalität für eine Reparatur. Es werden lediglich die Integritätsprüfungen durchgeführt, die in *nLevel* angegeben sind.
- SET TABLEVALIDATE betrifft keine temporären Dateien oder Cursor.
- Ist eine Tabelle exklusiv geöffnet, ignoriert Visual FoxPro die Einstellung des vierten Bits (Prüfen des Headers der Tabelle (.dbf) vor einer Einfügeoperation...=).

## Aktualisierungen an IntelliSense

Um Ihre IntelliSense-Datei zu aktualisieren, können Sie das folgende Skript einsetzen, indem Sie den Code in eine neue Programmdatei (.prg) kopieren und die Datei ausführen:

```
* Script for updating IntelliSense FoxCode table with new SET
* TABLEVALIDATE settings. Make sure the script appears in your PRG file as
* it does below. Copying the text from your Web browser might add extra
* empty lines (spacing within TEXT...ENDTEXT is critical).
LOCAL lcSaveArea,lcTipStr
lcSaveArea=SELECT()
SELECT 0
USE (_FOXCODE) SHARED
TEXT TO lcTipStr NOSHOW
SET TABLEVALIDATE TO [nLevel]

nLevel Values (additive):

1 - When a table is opened, the record count is validated. *
2 - When records are appended (or inserted) and written to disk, the record
count is validated.
4 - Suppress "File is in use" error (Error 3) if header cannot be locked for
table open operation. **
8 - Check DBF header before append is saved to disk and the header is modified.
This affects insert operations.

Note: No table validation is performed for value of 0.

* The table (dbf) header is locked during validation, which can impact
scalability for shared/multi-user scenarios.
** Requires nLevel value of 1 to also be set.

ENDTEXT

UPDATE (_FOXCODE) SET tip = lcTipStr WHERE UPPER(ALLTRIM(Expanded))=="SET
TABLEVALIDATE"
USE
SELECT (lcSaveArea)
```

## Neue Funktionalität des Task Pane Proxy Servers

Der Task Pane Manager enthält jetzt Optionen, um die Proxy Server-Einstellungen zu steuern, der von Task Panes benötigt wird, die Ihre Inhalte über XML Webdienste beziehen. Wenn Sie auf die Schaltfläche Optionen klicken und das Element Task Pane Manager auswählen, werden Sie eine neue Seite Proxy Server-Optionen bemerken. Diese Seite enthält die folgenden Optionen:

- **Keinen Proxy Server einsetzen.** Keine expliziten Proxy Server-Einstellungen für XML Webdienst-Aufrufe vornehmen, die für das Lesen der Inhalte verschiedener Task Panes vorgenommen werden, beispielsweise das Fenster Community.
- **Die folgenden Proxy Server-Einstellungen verwenden.** Ist dies ausgewählt, stehen die folgenden Optionen für die manuelle Steuerung der Einstellungen des Proxy Servers zur Verfügung:
  - **Adresse** Gibt die Adresse des Proxy Servers an, zum Beispiel <http://myproxyserver>, der für die Verbindung zum Internet eingesetzt wird.
  - **Port** Legt den Port des Proxy Servers fest, beispielsweise 80, der für die Verbindung zum Internet eingesetzt wird.

- **Anwender** Legt eine optionale Benutzernamen-Einstellung fest, wenn dieser erforderlich ist, um sich einzuloggen.
- **Passwort** Legt eine optionale Passwort-Einstellung fest, wenn diese erforderlich ist, um sich einzuloggen.

## Verschiedene Änderungen des Verhaltens

- Der Einsatz des Befehls APPEND FROM mit einem Autoinkrement-Feld generiert für dieses Feld jetzt neue Werte. Die neuen Werte werden automatisch sequentiell auf der Basis des letzten generierten Wertes generiert. Dies geschieht unabhängig davon, ob im Befehl APPEND FROM oder APPEND FROM ARRAY Werte für das Autoinkrement-Feld angegeben sind.
- Ein Befehl SEEK mit einem Parameter ORDER nutzt jetzt das vorgegebene Schlüsselwort ASC oder DESC für den Indextag. Das vorherige Verhalten war nicht kalkulierbar und fußte auf der vorherigen Einstellung des Schlüsselworts DESCENDING oder ASCENDING für den Befehl SEEK.
- Wird ein Tabellenfeld eines DBC mit einem Ausdruck für **Caption** auf ein Formular gezogen, enthält die resultierende Caption **Label** jetzt den folgenden Ausdruck:

```
=Iif(Used(<tablename>) and !Empty(Field(<fieldname>, <tablename>, 1)),Field(<fieldname>, <tablename>, 1), <fieldname>)
```

- In der originalen Version von Visual FoxPro 8.0 lautete der generierte Ausdruck:

```
=FIELD( <tablename>, <fieldname>, 1)
```